

OPTIMASI FUNGSI KEANGGOTAAN FUZZY TSUKAMOTO DENGAN ALGORITME GENETIKA PADA PERAMALAN HARGA EMAS UNTUK STOCK TRADING

SKRIPSI

Untuk memenuhi sebagian persyaratan
Memperoleh gelar Sarjana Komputer

Disusun oleh:
Ficry Agam Fathurrachman
NIM: 145150200111121



PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2018

PENGESAHAN

Optimasi Fungsi Keanggotaan Fuzzy Tsukamoto dengan Algoritme Genetika pada
Peramalan Harga Emas untuk *Stock Trading*

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh :
Ficry Agam Fathurrachman
NIM: 145150200111121

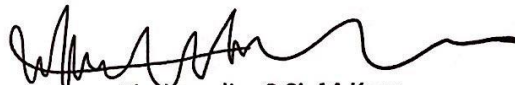
Skripsi ini telah diuji dan dinyatakan lulus pada
20 Desember 2018
Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II



Fitra A. Bachtiar, Dr. Eng., S.T, M.Eng
NIK: 201201 840628 1 001



Imam Cholissodin, S.Si, M.Kom
NIK: 201201 850719 1 001

Mengetahui

Ketua Jurusan Teknik Informatika





H. Satrio Nugroho, S.T, M.T, Ph.D
NID: 19710518 200312 1 001

IDENTITAS TIM PENGUJI

PENGUJI 1

Sigit Adinugroho, S.Kom., M.Sc

NIK/NIP: 201607 880701 1 000

Ir. Sutrisno, M.T

NIK/NIP: 195703251987011001




PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 16 Juli 2018



Agam Fathurrachman
NIM: 145150200111121

ABSTRAK

Investor dan pedagang saham memerlukan pengetahuan tentang peramalan kapan harga emas akan naik atau akan turun agar dapat meminimalkan risiko dalam berinvestasi. Peramalan ini membutuhkan sebuah metode yang tepat agar dapat memberikan hasil yang baik. FIS Tsukamoto digunakan untuk meramalkan harga emas berdasarkan data kurs mata uang yang ada. Parameter yang digunakan oleh FIS Tsukamoto adalah kurs mata uang USD/GBP, CHF/USD, JPY/USD, EUR/USD berdasarkan tiga hari sebelumnya dan harga emas berdasarkan satu hari sebelumnya. Untuk memaksimalkan performa FIS Tsukamoto maka fungsi keanggotaan FIS Tsukamoto akan dioptimasi menggunakan Algoritme Genetika. Representasi kromosom yang digunakan adalah *real-coded* dengan tipe data *double*. Pada reproduksi metode *crossover* yang digunakan adalah *one-cut point*, sedangkan untuk metode mutasi yang digunakan adalah *random mutation*. Pada proses seleksi, metode yang digunakan adalah *elitism selection* untuk mendapatkan individu yang terbaik. Berdasarkan pengujian parameter yang dilakukan dengan 10 kali percobaan setiap parameter didapatkan ukuran populasi terbaik sebanyak 180, kombinasi $cr=0.9$ dan $mr=0.1$, serta jumlah generasi terbaik sebesar 325 nilai *fitness* terbaik sebesar 8,6870. Nilai Root-Mean Squared Error (RMSE) yang didapatkan sebelum optimasi adalah 13,3611, sedangkan setelah dilakukan optimasi didapatkan nilai RMSE yang lebih kecil yaitu sebesar 12,5801. Hasil tersebut menunjukkan adanya peningkatan nilai akurasi pada FIS Tsukamoto setelah dioptimasi dengan menggunakan Algoritme Genetika.

Kata Kunci: Optimasi, Peramalan, Harga Emas, Algoritme Genetika, FIS Tsukamoto

ABSTRACT

Investors and stock traders need knowledge of forecasting when the price of gold will rise or will decline to minimize the risk in investing. This forecasting requires an appropriate method in order to give good results. FIS Tsukamoto is used to forecast the price of gold based on existing exchange rate data. The parameters used by Tsukamoto FIS are the currency rates of USD / GBP, CHF / USD, JPY / USD, EUR / USD based on the previous three days and the price of gold based on the previous day. To maximize Tsukamoto FIS performance, Tsukamoto FIS membership function will be optimized using Genetic Algorithm. The chromosome representation used is real-coded with a double data type. The reproduction of the crossover method used is one-cut point, while the mutation method used is random mutation. In the selection process, the method used is elitism selection to get the best individuals. Based on parameter testing carried out with 10 experiments each parameter, the best population size is 180, combination of $cr = 0.9$ and $mr = 0.1$, and the best number of generations is 325, the best fitness value is 8.6972. The Root-Mean Squared Error (RMSE) value obtained before optimization is 13.3611, while after optimization it is obtained that the smaller RMSE value is 12.5801. These results indicate an increase in the value of accuracy in Tsukamoto FIS after being optimized using Genetic Algorithm.

Keywords: Optimazion, Forecasting, Gold Prices, Genetic Algorithm, Tsukamoto FIS



DAFTAR ISI

PERSETUJUAN	i
PERNYATAAN ORISINALITAS	ii
KATA PENGANTAR.....	iii
ABSTRAK.....	v
ABSTRACT.....	vi
DAFTAR ISI.....	vii
DAFTAR TABEL	x
DAFTAR GAMBAR.....	xi
DAFTAR KODE PROGRAM	xiii
DAFTAR LAMPIRAN	xiv
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	3
1.3 Tujuan	3
1.4 Manfaat.....	3
1.5 Batasan Masalah.....	3
1.6 Sistematika Pembahasan	3
BAB 2 KAJIAN PUSTAKA	5
2.1 Kajian Pustaka	5
2.2 Investasi Emas.....	6
2.3 Perdagangan Saham (<i>Stock Trading</i>)	7
2.4 Fuzzy Inference System (FIS) Tsukamoto.....	8
2.4.2 Algoritme Genetika	11
2.4.3 Siklus Algoritme Genetika	12
2.5 Evaluasi Sistem.....	15
BAB 3 METODOLOGI	16
3.1 Tipe Penelitian	16
3.2 Strategi Penelitian.....	16
3.3 Teknik Pengumpulan Data	16
3.4 Perancangan Sistem.....	17
3.5 Pengujian dan Analisis	17
3.6 Penarikan Kesimpulan dan Saran.....	18
BAB 4 PERANCANGAN ALGORITME	19
4.1 Deskripsi Masalah	19

4.2 Deskripsi Umum Sistem	19
4.3 Diagram Alir Sistem.....	19
4.4 Manualisasi Perhitungan Data	31
4.4.1 FIS Tsukamoto.....	33
4.4.2 Menghitung Nilai Error.....	40
4.4.3 Algoritme Genetika	40
4.5 Analisis Kebutuhan Perangkat Lunak.....	46
4.5.1 Identifikasi Pengguna	46
4.5.2 Analisis Kebutuhan Masukan	46
4.5.3 Analisis Kebutuhan Keluaran	46
4.6 Perancangan Sistem.....	47
4.7 Perancangan Antarmuka.....	47
4.7.1 Halaman Data.....	47
4.7.2 Halaman Optimasi.....	48
4.7.3 Halaman Perbandingan.....	49
4.7.4 Halaman Peramalan	49
4.8 Perancangan Pengujian.....	50
4.8.1 Pengujian Populasi	50
4.8.2 Pengujian Nilai Crossover rate dan Mutation rate	51
4.8.3 Pengujian Konvergensi.....	51
4.8.4 Pengujian RMSE (Root Mean Squared Error).....	51
BAB 5 IMPLEMENTASI.....	53
5.1 Spesifikasi Perangkat.....	53
5.1.1 Spesifikasi Perangkat Keras.....	53
5.1.2 Spesifikasi Perangkat Lunak	53
5.2 Implementasi Algoritme	53
5.2.1 Implementasi Proses Inisialisasi Kromosom	53
5.2.2 Implementasi Proses <i>Crossover</i>	54
5.2.3 Implementasi Proses Mutasi.....	55
5.2.4 Implementasi Proses Evaluasi.....	56
5.2.5 Implementasi Proses Seleksi.....	57
5.2.6 Implementasi Perhitungan Nilai Keanggotaan	57
5.2.7 Implementasi Perhitungan α -Predikat.....	59
5.2.8 Implementasi Perhitungan Nilai Z.....	61
5.2.9 Implementasi Proses Defuzzifikasi dan Perhitungan Fitness.....	62
5.3 Implementasi Antarmuka	63
5.3.1 Implementasi Antarmuka Laman Data Latih	63

5.3.2 Implementasi Anatrmuka Laman Data Uji	64
5.3.3 Implementasi Antarmuka Laman Optimasi	64
5.3.4 Implementasi Antarmuka Laman Perbandingan	65
5.3.5 Implementasi Antarmuka Laman Peramalan	65
BAB 6 PENGUJIAN DAN ANALISIS.....	67
6.1 Pengujian Ukuran Populasi dan Analisis	67
6.2 Pengujian Kombinasi <i>cr</i> dan <i>mr</i> dan Analisis	68
6.3 Pengujian Konvergensi dan Analisis.....	70
6.4 Pengujian Dengan Parameter Terbaik	72
6.5 Analisis Global.....	76
BAB 7 PENUTUP	78
7.1 Kesimpulan.....	78
7.2 Saran	78
DAFTAR PUSTAKA	80
LAMPIRAN A DATA.....	83
A.1 DATA LATIH.....	83
A.2 DATA UJI	105
LAMPIRAN B PENGUJIAN	108
B.1 PENGUJIAN POPULASI.....	108
B.2 PENGUJIAN KOMBINASI CR DAN MR.....	109
B.3 PENGUJIAN KONVERGENSI	109

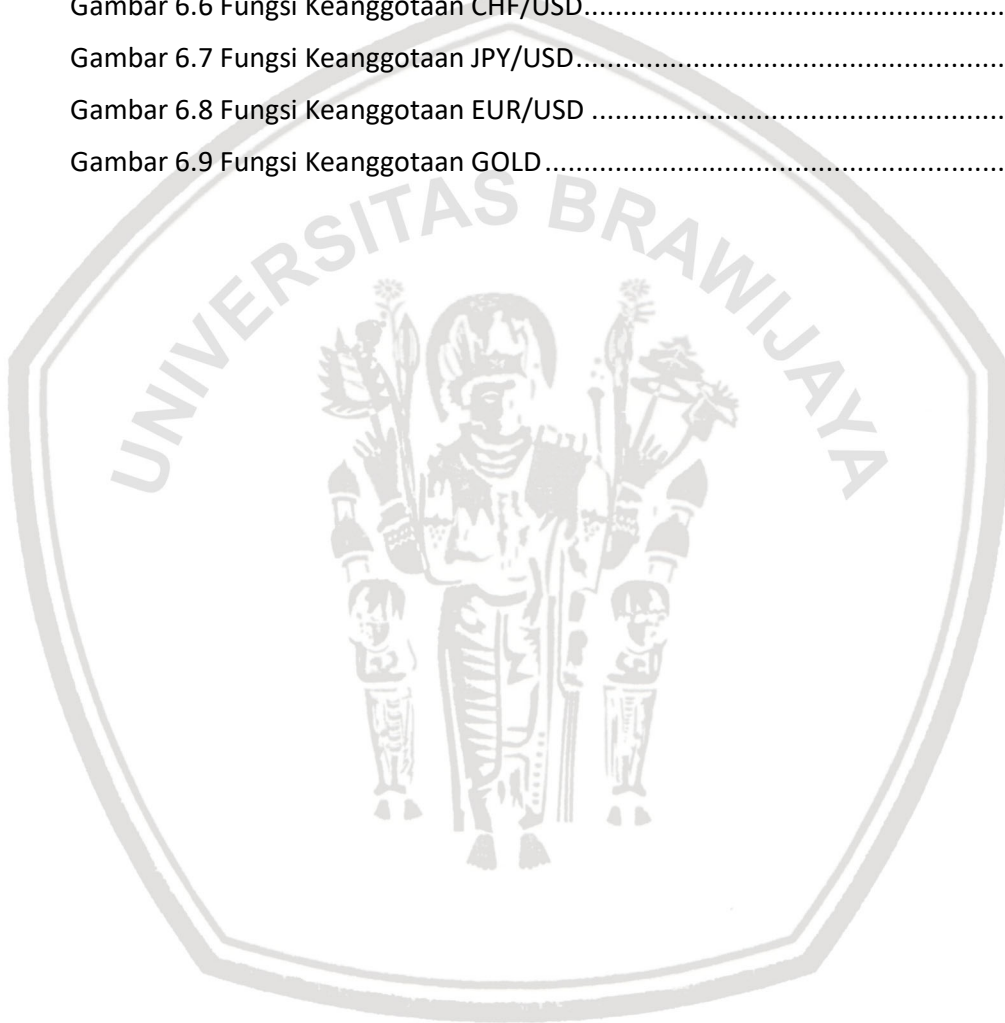
DAFTAR TABEL

Tabel 4.1 Data Latih	31
Tabel 4.2 Data Uji.....	32
Tabel 4.9 Representasi Kromosom	40
Tabel 4.10 Crossover <i>One Cut Point</i>	41
Tabel 4.11 Mutasi	42
Tabel 4.12 Nilai Evaluasi.....	42
Tabel 4.13 Seleksi berdasarkan <i>fitness</i>	43
Tabel 4.14 Seleksi pada Iterasi kedua	43
Tabel 4.15 Gen dari individu yang terpilih	43
Tabel 4.16 Perancangan Pengujian Populasi	50
Tabel 4.17 Perancangan Pengujian Kombinasi Nilai <i>cr</i> dan <i>mr</i>	51
Tabel 4.18 Perancangan Pengujian Konvergensi	51
Tabel 6.1 Pengujian Populasi.....	67
Tabel 6.2 Pengujian Kombinasi <i>cr</i> dan <i>mr</i>	69
Tabel 6.3 Pengujian Konvergensi.....	71
Tabel 6.4 Pengujian Parameter Terbaik.....	72
Tabel 6.5 Individu Terbaik	73
Tabel 6.6 Perbandingan FIS dan GA-FIS	76

DAFTAR GAMBAR

Gambar 2.1 Arsitektur Fuzzy Logic Controller.....	8
Gambar 2.2 Skema penalaran FIS Tsukamoto	9
Gambar 2.3 Representasi Kurva Linear Naik.....	10
Gambar 2.4 Representasi Kurva Linear Turun	10
Gambar 2.5 Representasi Kurva Segitiga	10
Gambar 2.6 Proses <i>crossover</i> dengan satu <i>crossover point</i>	13
Gambar 4.1 Diagram Alir Sistem Optimasi Secara Umum.....	20
Gambar 4.2 Diagram Alir Sistem Peramalan Secara Umum	21
Gambar 4.3 Diagram Alir Metode FIS Tsukamoto.....	22
Gambar 4.4 Diagram Alir Inferensi	23
Gambar 4.5 Diagram Alir Proses Defuzzifikasi	24
Gambar 4.6 Diagram Alir Metode Algoritma Genetika	25
Gambar 4.7 Diagram Alir Inisialisasi Kromosom	26
Gambar 4.8 Diagram Alir <i>Crossover</i>	28
Gambar 4.9 Proses Mutasi	29
Gambar 4.10 Diagram Alir Evaluasi	30
Gambar 4.11 Diagram Alir Seleksi	31
Gambar 4.12 Fungsi Keanggotaan USD/GBP	33
Gambar 4.13 Fungsi Keanggotaan CHF/USD.....	34
Gambar 4.14 Fungsi Keanggotaan JPY/USD.....	34
Gambar 4.15 Fungsi Keanggotaan EUR/USD	34
Gambar 4.16 Fungsi Keanggotaan GOLD	35
Gambar 4.17 Fungsi Keanggotaan USD/GBP	44
Gambar 4.18 Fungsi Keanggotaan CHF/USD.....	44
Gambar 4.19 Fungsi Keanggotaan JPY/USD.....	45
Gambar 4.20 Fungsi Keanggotaan EUR/USD	45
Gambar 4.21 Fungsi Keanggotaan GOLD	46
Gambar 4.22 Perancangan Halaman Data	48
Gambar 4.23 Perancangan Halaman Optimasi	48
Gambar 4.24 Perancangan Halaman Perhitungan	49
Gambar 4.25 Perancangan Halaman Peramalan	50
Gambar 5.1 Antarmuka Laman Data Latih.....	64

Gambar 5.2 Antarmuka Laman Data Uji	64
Gambar 5.3 Implementasi Laman Optimasi	65
Gambar 5.4 Implementasi Laman Perbandingan.....	65
Gambar 5.5 Implementasi Laman Peramalan.....	66
Gambar 6.1 Pengujian Ukuran Populasi	68
Gambar 6.2 Pengujian Kombinasi cr dan mr	69
Gambar 6.3 Pengujian Konvergensi.....	71
Gambar 6.4 Pengujian dengan Parameter Terbaik	73
Gambar 6.5 Fungsi Keanggotaan USD/GBP	74
Gambar 6.6 Fungsi Keanggotaan CHF/USD.....	74
Gambar 6.7 Fungsi Keanggotaan JPY/USD.....	75
Gambar 6.8 Fungsi Keanggotaan EUR/USD	75
Gambar 6.9 Fungsi Keanggotaan GOLD.....	76



BAB 1 PENDAHULUAN

1.1 Latar Belakang

Investasi adalah komitmen terhadap sumber daya yang ada pada masa sekarang dengan tujuan untuk mendapatkan keuntungan di kemudian hari (Halim, 2005). Secara sederhana investasi dapat dikatakan membeli dengan harga rendah kemudian menjualnya kembali dengan harga yang lebih tinggi di masa depan. Berdasarkan definisi ini, waktu dan masa depan adalah dua elemen penting dalam investasi. Maka dari itu, informasi tentang tingkat kepastian status investasi di masa mendatang akan sangat membantu investor.

Emas berperan penting dalam ekonomi dunia dan memiliki kedudukan yang baik di dalam bidang keuangan dan makroekonomi (Pierdzioch, et al., 2014). Emas memiliki nilai moneter dan dicari oleh bank pusat untuk dijadikan sebagai devisa negara yang mana bisa digunakan untuk berbagai tujuan (Gupta, et al., 2014). Pada bidang keuangan masa kini, emas digunakan untuk menjaga nilai ketika inflasi dan mengamankan aset ketika krisis (Hassani, et al., 2015). Hal ini disebabkan karena harga emas yang selalu stabil. Harga dan produksi emas berbeda dari sebagian besar mineral lain. Pada krisis keuangan tahun 2008, harga emas naik sebesar 6% sementara sebagian besar harga mineral dan ekuitas lain turun sebesar 40% (Shafiee & Topal, 2010). Menurut data yang diperoleh dari PT. Monex Investindo Futures pada bulan Juli 2008 harga emas per *troy ounce* menyentuh angka US\$ 914,40 dan pada bulan Januari 2014 harga emas per *troy ounce* mencapai angka US\$ 1245,05 (Kusumadewi, 2014). Hal inilah yang kemudian membuat minat beli investor dan pedagang terhadap emas menjadi sangat tinggi.

Transaksi emas adalah bentuk perdagangan yang melibatkan pasar utama dunia. Mata uang yang diperdagangkan atas dasar global dan nilainya selalu berubah di setiap detiknya. Perdagangan emas dalam perdagangan berjangka adalah perdagangan yang melakukan transaksi melalui kontrak beli atau kontrak jual dari barang dagangan yang diperjualbelikan di bursa. Pengambilan keputusan dalam setiap transaksi dapat menentukan jumlah keuntungan ataupun kerugian yang didapat ketika melakukan pembelian atau penjualan. Dalam dunia investasi, resiko kerugian yang tidak diharapkan dalam pengambilan keputusan memiliki kemungkinan yang besar untuk terjadi. Karena pada dasarnya tidak ada investasi yang tidak beresiko (Indriani & Harjahdi, 2013).

Untuk meminimalisir resiko yang didapat maka investor dan pedagang saham memerlukan pengetahuan tentang peramalan harga emas untuk mengetahui kapan harga emas akan naik atau akan turun. Peramalan adalah sebuah proses untuk memperkirakan besar atau jumlah sesuatu di masa depan berdasarkan data yang didapat dari masa lalu (*time series*) yang dianalisis dengan metode ilmiah seperti metode statistika (Sudjana, 1996). Peramalan harga emas memiliki tujuan untuk mengetahui peluang harga emas di masa depan sehingga dapat menjadi acuan investor untuk mengetahui perubahan harga emas (Kusumadewi, 2014).

Salah satu metode yang dapat digunakan untuk meramal harga emas adalah metode *fuzzy*. Metode ini banyak diterapkan dalam melakukan peramalan.

Sihananto (2017) menggunakan metode *Fuzzy Inference System* (FIS) yang dikombinasikan dengan *Evolution Strategies* (ES) untuk meramalkan harga emas dan mendapatkan nilai *Mean Absolute Error* (MAE) sebesar 2,734. Sukandy (2014) juga menggunakan metode FIS untuk memprediksi jumlah produksi minyak sawit dan diperoleh nilai produksi yang memenuhi permintaan sebesar 86,67% dari 30 data yang dipilih secara acak. Hasibuan (2016) menggunakan metode *fuzzy* untuk memprediksi harga kelapa sawit dan menghasilkan nilai MSE untuk data training sebesar 1114.229, nilai MAPE sebesar 2.9217. Nilai MSE untuk data testing sebesar 2443.776 dan nilai MAPE sebesar 2.719. Dharmawati & Aprilianto (2014) menggunakan metode Tsukamoto untuk memprediksi nilai tukar rupiah dan menghasilkan nilai simpangan rata-rata (AFER) untuk Dolar AS sebesar 10,74%, *Poundsterling* sebesar 13,26%, dan *Euro* sebesar 13,54%. Penelitian-penelitian ini membuktikan bahwa metode *fuzzy* mampu melakukan prediksi atau peramalan untuk kasus yang berbeda-beda karena metode *fuzzy* sangat fleksibel terhadap perubahan dan ketidakpastian masalah (Kusumadewi & Purnomo, 2010).

Penggunaan Logika *fuzzy* pada permasalahan yang sangat kompleks membuat performanya kadangkala dirasa masih kurang optimal. Permasalahan yang umumnya dijumpai adalah sulitnya menemukan *fuzzy set* dan *rule fuzzy* untuk menemukan solusi pada permasalahan (Alharbi & Tchier, 2015). Menurut Esmin (2007) fungsi keanggotaan dan *rule fuzzy* akan mempengaruhi performa metode *fuzzy* tersebut, sehingga metode *fuzzy* ini kemudian dapat dikembangkan dengan cara mengubah fungsi keanggotaannya.

Pada penelitian yang dilakukan Sihananto (2017) dalam meramalkan harga emas hanya mengoptimasi *rule fuzzy* saja, sedangkan fungsi keanggotaan didapatkan dari pakar. Maka dari itu pada penelitian ini akan mengoptimasi fungsi keanggotaan *fuzzy* agar mengurangi ketergantungan kepada pakar sehingga dapat memberikan solusi yang lebih optimal dan memberikan nilai *error* yang lebih rendah. Untuk memaksimalkan peramalan dengan metode *fuzzy* tersebut, maka fungsi keanggotaan *fuzzy* akan dioptimasi menggunakan Algoritme Genetika. Optimasi fungsi keanggotaan *fuzzy* dengan Algoritme Genetika ini sudah banyak dilakukan sebelumnya. Misalkan penelitian yang dilakukan oleh Kurnianingtyas (2017) yang menggunakan Algoritme Genetika untuk mengoptimasi derajat keanggotaan FIS Tsukamoto untuk mendiagnosis penyakit pada sapi potong. Fitri (2017) mengoptimasi fungsi keanggotaan FIS Tsukamoto dengan menggunakan Algoritme Genetika untuk menentukan prioritas penerima zakat. Armanda (2016) juga menggunakan metode Algoritme Genetika dalam mengoptimasi FIS Tsukamoto untuk meramalkan permintaan barang semen, dimana pada penelitian ini menunjukkan bahwa hasil prediksi setelah dioptimasi menjadi lebih baik dibandingkan dengan sebelum dioptimasi. Ketiga penelitian ini membuktikan bahwa Algoritme Genetika mampu mengoptimasi fungsi keanggotaan FIS Tsukamoto dalam kasus yang berbeda-beda.

Berdasarkan uraian diatas, maka penelitian ini mengusulkan model FIS Tsukamoto yang dikombinasikan dengan Algoritme Genetika untuk data *time series* dan peranannya dalam meramalkan harga emas. Penelitian ini akan mengimplementasikan metode Algoritme Genetika untuk mengoptimasi fungsi keanggotaan FIS Tsukamoto dalam peramalan harga emas. Kemudian akan dilihat

bagaimana pengaruh proses optimasi ini terhadap nilai *error* yang dihasilkan oleh sistem. Penelitian ini berfokus mencari fungsi keanggotaan yang mendekati optimal berdasarkan parameter-parameter yang ada. Fungsi keanggotaan ini kemudian dapat digunakan untuk meramalkan harga emas. Diharapkan dari penelitian ini dapat membantu para investor maupun pedagang saham sebagai pertimbangan dalam berinvestasi.

1.2 Rumusan Masalah

Berdasarkan latar belakang diatas terdapat dua rumusan masalah :

1. Bagaimana mengaplikasikan metode Algoritma Genetika untuk mengoptimasi fungsi keanggotaan dari FIS Tsukamoto dalam meramalkan harga emas?
2. Berapa nilai *error rate* yang dihasilkan oleh sistem dengan menggunakan metode Tsukamoto untuk meramalkan harga emas setelah dioptimasi dengan menggunakan metode Algoritme Genetika?

1.3 Tujuan

1. Mengimplementasikan metode Algoritme Genetika untuk mengoptimasi fungsi keanggotaan FIS Tsukamoto dalam meramalkan harga emas.
2. Mengetahui *error rate* yang dihasilkan oleh sistem yang menggunakan metode Tsukamoto untuk meramalkan harga emas setelah dioptimasi dengan menggunakan metode Algoritme Genetika.

1.4 Manfaat

1. Membantu para pedagang saham yang ingin membeli emas untuk mengetahui pergerakan harga emas ketika naik atau turun.
2. Membantu orang-orang yang ingin berinvestasi emas untuk memberikan gambaran pergerakan harga emas setiap harinya.

1.5 Batasan Masalah

Guna mencegah meluasnya cakupan permasalahan yang akan dibahas dalam studi ini dan untuk membuat studi ini lebih terarah, maka dilakukan pembatasan masalah adalah sebagai berikut:

1. Data yang digunakan menggunakan data yang diambil dari Quandl, World Gold Council, Forex Trading dan US Energy Information Administration.
2. Data yang digunakan adalah pergerakan kurs mata uang dolar Amerika terhadap mata uang negara lain, antara lain USD/GBP, CHF/USD, JPY/USD, EUR/USD dan harga emas.
3. Perancangan program aplikasi sistem optimasi fungsi keanggotaan dalam peramalan harga emas ini menggunakan bahasa pemrograman Java.

1.6 Sistematika Pembahasan

Adapun pada penyusunan skripsi terdapat penulisan terstruktur yakni :

BAB 1 PENDAHULUAN

Merupakan bab yang berisi bagian awal seperti latar belakang, rumusan masalah, tujuan penelitian, batasan masalah, manfaat penelitian dan sistematika

pembahasan optimasi fungsi keanggotaan FIS Tsukamoto dengan menggunakan Algoritme Genetika pada peramalan harga emas untuk *Stock Trading*.

BAB 2 LANDASAN KEPUSTAKAAN

Merupakan bab yang berisi dasar teori dan juga tabel kajian pustaka terkait pada optimasi fungsi keanggotaan FIS Tsukamoto dengan menggunakan Algoritme Genetika pada peramalan harga emas untuk *Stock Trading*.

BAB 3 METODOLOGI PENELITIAN

Merupakan bagian yang berisi pengumpulan data, analisa kebutuhan, perancangan dalam sistem, implementasi dan kesimpulan optimasi fungsi keanggotaan FIS Tsukamoto dengan menggunakan Algoritme Genetika dalam peramalan harga emas.

BAB 4 ANALISIS DAN PERANCANGAN

Pada bab ini berisikan perancangan yang terdiri perancangan sistem untuk hasil optimasi model metode FIS Tsukamoto, perancangan antarmuka, serta perancangan uji coba dan evaluasi.

BAB 5 IMPLEMENTASI

Pada bab ini berisi pembahasan proses implementasi, batasan-batasan implementasi, serta algoritme yang digunakan dalam sistem.

BAB 6 PENGUJIAN

Pada bab ini memuat pengujian sistem dan analisis hasil pengujian tentang penggunaan metode Algoritme Genetika dan FIS Tsukamoto.

BAB 7 KESIMPULAN

Pada bab ini memuat kesimpulan dari hasil penelitian dan saran untuk pengembangan sistem selanjutnya agar lebih baik.

BAB 2 KAJIAN PUSTAKA

Bab ini berisikan tentang landasan pustaka dan beberapa dasar teori yang berhubungan dengan penelitian skripsi ini secara umum. Landasan pustaka membahas teori-teori yang digunakan untuk meramalkan harga emas dengan menggunakan metode Tsukamoto untuk *Stock Trading* dan juga optimasi yang dilakukan dengan menggunakan metode Algoritme Genetika.

2.1 Kajian Pustaka

Penelitian tentang peramalan harga emas sebelumnya adalah Gold Price Movement Forecasting using Hybrid ES-FIS oleh Andreas Nugroho Sihananto dan Fitra A. Bachtiar (2017). Penelitian ini menggunakan metode *Evolution Strategies* (ES) untuk membangkitkan *rules* fuzzy-nya dan kemudian perhitungan peramalan menggunakan metode FIS. Pada penelitian ini ditujukan untuk mengembangkan dua buah metode yaitu ES untuk membangkitkan *rules* dan perhitungan menggunakan metode FIS berdasarkan *rules* yang ada guna menentukan peramalan harga emas. Evolution Strategies memiliki ciri yaitu penggunaan vektor bilangan pecahan (*real-vector*) sebagai representasi solusi. Berbeda dengan Algoritme Genetika yang menggunakan *crossover* sebagai operator reproduksi utama dan mutasi sebagai penunjang, pada ES lebih menekankan pada operator mutasi (Mahmudy, 2015). Pengujian sistem dilakukan dengan melakukan proses terhadap data yang didapatkan dari tanggal 25 Agustus 2011 sampai 30 Desember 2016 Metode ES-FIS kemudian menggunakan parameter yang sudah ditentukan. Hasilnya kemudian dibandingkan dengan metode FIS Mamdani biasa dan kemudian dievaluasi dengan MAE. Dan hasil yang dicapai pada pengembangan metode ini adalah metode ES-FIS memiliki kemiripan yang lebih tinggi dengan data asli dibandingkan dengan metode FIS Mamdani. Hal ini disebabkan karena ES memiliki kemampuan menyesuaikan sendiri. Perbedaan hasil ES-FIS dan FIS Mamdani tidak terlalu signifikan. Namun pada prakteknya, perbedaan kecil ini dapat berdampak besar pada untung dan rugi.

Optimasi fungsi keanggotaan FIS Tsukamoto dengan menggunakan Algoritme Genetika dalam kasus peramalan dilakukan oleh Armanda (2016) untuk meramalkan jumlah permintaan barang. Penelitian ini membangkitkan *rule* FIS Tsukamoto dengan menggunakan Algoritme Genetika. Terdapat 4 kriteria dalam perhitungan FIS, yang pertama permintaan berdasarkan 3 minggu sebelumnya, kemudian permintaan berdasarkan 2 minggu sebelumnya, kemudian permintaan berdasarkan 1 minggu sebelumnya, dan permintaan berdasarkan 1 minggu selanjutnya. Pada fungsi keanggotaan terdapat 3 kriteria yaitu Rendah, Sedang dan Tinggi. Algoritme Genetika kemudian digunakan untuk menentukan batas fungsi keanggotaan FIS tersebut. Pada representasi kromosom dibangkitkan nilai acak sebagai batasan-batasan FIS Tsukamoto. Kemudian akan dicari nilai *Root-Mean Square Error* (RMSE) dari peramalan dengan setiap batasan-batasan tersebut. Nilai RMSE ini kemudian digunakan untuk mendapatkan nilai *fitness*. Dari penelitian ini didapatkan bahwa setelah dioptimasi, nilai *error* yang dihasilkan mengalami penurunan. Nilai *error* sebelum dioptimasi adalah sebesar 5770,44. Setelah dioptimasi nilai *error* yang didapatkan menurun menjadi 1202,42.

Penelitian menggunakan metode FIS Tsukamoto juga dilakukan oleh Surya (2019) untuk meramalkan produksi roti. Penelitian tersebut dilakukan dengan menggunakan 3 jenis roti yaitu roti tawar, roti *cake*, dan roti manis. Kemudian data tersebut diolah dengan menggunakan metode FIS Tsukamoto dengan menentukan himpunan *input* dan himpunan *output* untuk data uji. Selanjutnya membuat himpunan *fuzzy* serta menentukan fungsi keanggotaannya yang didapatkan dari pakar. Kemudian dilakukan pengujian dengan menggunakan 20 data, 40 data dan 60 data. Setelah permodelan dilakukan, kemudian menentukan nilai *Root-Mean Square Error* (RMSE). Langkah ini bertujuan untuk mengevaluasi model FIS Tsukamoto. Penelitian ini menghasilkan nilai RMSE untuk roti manis sebesar 69,84, roti *cake* sebesar 9,95, dan roti tawar sebesar 4,2. Dari penelitian tersebut disimpulkan bahwa peramalan untuk roti manis dianggap buruk, namun peramalan untuk roti tawar dan *cake* dapat dikatakan cukup baik

2.2 Investasi Emas

Pada dasarnya orang memilih untuk berinvestasi atau berdagang dalam bentuk emas untuk mendapatkan profit. Investasi emas dibedakan menjadi 2 yaitu emas batangan dan saham emas (sertifikat). Keduanya memiliki peluang investasi dan risikonya sendiri. Berinvestasi dalam bentuk emas batangan membutuhkan biaya untuk menyewa *safe deposit box* dan memungkinkan resiko lebih besar daripada berinvestasi emas dalam bentuk saham. Dalam berinvestasi emas berbentuk saham, investor hanya perlu keahlian membaca bursa saham (Kusumadewi, 2014).

Emas memiliki persediaan yang terbatas dan sulit didapat, sementara persediaan terhadap emas terus meningkat. Akibatnya harga emas cenderung mengalami kenaikan dari tahun ke tahun. Pada kenyataannya, harga emas tidak hanya bergantung pada permintaan dan persediaan. Harga emas juga dipengaruhi oleh situasi perekonomian secara keseluruhan (Apriyanti, 2012). Faktor faktor yang memengaruhi harga emas adalah sebagai berikut.

1. Inflasi

Inflasi terjadi jika nilai mata uang mengalami penurunan sehingga harga barang di pasaran mengalami kenaikan. Hal ini didorong oleh meningkatnya permintaan barang dan jasa yang kemudian diperparah dengan tersendatnya distribusi barang. Kondisi inflasi seperti inilah yang mendorong naiknya harga emas.

2. Krisis Finansial

Saat terjadi krisis finansial, orang lebih jeli dalam berinvestasi. Orang akan mencari keamanan dalam berinvestasi, sehingga mereka akan memilih investasi yang memberikan keuntungan. Dalam kasus ini berinvestasi dengan emas.

3. Naiknya permintaan

Harga emas bisa naik akibat permintaan yang sangat banyak. Ketika permintaan terus menerus naik, maka emas akan menjadi langka. Kelangkaan ini yang kemudian menyebabkan harga emas menjadi naik.

4. Kurs Dolar

Karena harga emas dihitung berdasarkan kurs dolar, maka jika dolar mengalami kenaikan maka harga emas juga akan ikut naik.

5. Harga minyak

Korelasi antara emas dan minyak adalah berbanding lurus. Jika harga minyak melonjak, maka produksi emas akan menurun karena biaya produksi semakin mahal. Penambangan emas sangatlah bergantung pada minyak untuk operasionalnya. Pada situasi lain, ketika harga minyak naik maka kinerja perusahaan akan menurun. Produksi pun menjadi menurun dan bisa jadi tidak mencapai target. Hal ini menyebabkan banyak investor dari perusahaan tersebut melepas kepemilikan sahamnya sehingga harga saham perusahaan itu merosot. Para investor pun mencari alternatif lain dalam berinvestasi, salah satunya emas. Permintaan yang tinggi terhadap emas disaat mereka melepas kepemilikan sahamnya menyebabkan harga emas menjadi naik.

6. Situasi politik dunia

Apabila kondisi politik dan ekonomi kacau balau, maka harga emas akan melonjak. Hal ini disebabkan karena emas seringkali dianggap sebagai penyelamat saat terjadinya gejolak ekonomi atau politik. Saat terjadi krisis atau perang, biasanya harga emas akan melonjak naik.

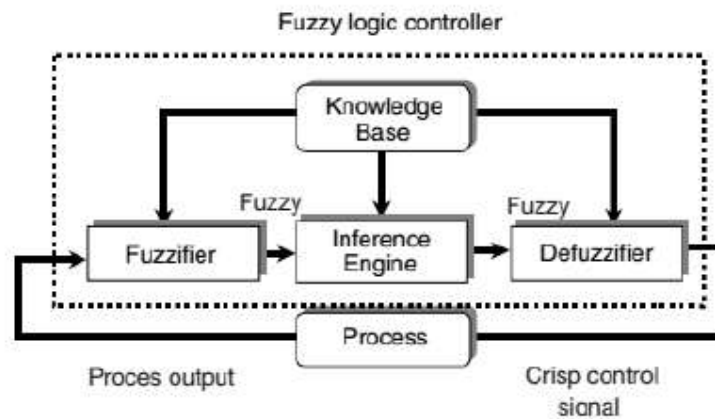
2.3 Perdagangan Saham (*Stock Trading*)

Perdagangan saham adalah kegiatan membeli atau menjual saham. Perdagangan dapat dilakukan dengan cepat melalui panggilan telepon atau menggunakan internet. Orang-orang sebelumnya berdagang saham menggunakan surat, namun karena kemajuan teknologi sekarang orang-orang dapat berdagang saham secara digital (Ni, 2006). Para investor saham dapat menghasilkan uang dari saham melalui apresiasi modal atau gratifikasi. Gratifikasi ini dibayarkan oleh perusahaan setiap 3 bulan.

Pada perdagangan di pasar saham terdapat 3 metode, yaitu metode perdagangan jangka panjang, perdagangan jangka menengah dan perdagangan jangka pendek. Perdagangan jangka pendek biasanya dilakukan kurang dari 1-2 tahun. Pedagang saham jangka pendek membeli dan menjual saham dengan cepat, jadi keuntungan dan kerugian yang didapatkan tidak banyak. Perdagangan jangka menengah dilakukan dalam rentang waktu 5 tahun. Para investor yang melakukan perdagangan jangka menengah biasanya memilih perusahaan yang besar dan terpercaya karena biasanya harga saham perusahaan tersebut lebih stabil. Perdagangan jangka panjang dilakukan selama lebih dari 5 tahun. Dalam aspek keuntungan, investasi saham jangka panjang dapat mengalahkan investasi keuangan seperti obligasi atau investasi bank dalam periode 10 tahun. Meskipun, prospeknya sangat bagus para investor harus jeli dalam memilih saham yang tepat, jika tidak investor akan kehilangan uang apabila perusahaan yang diinvestasikan bangkrut setelah beberapa tahun (Ni, 2006).

2.4 Fuzzy Inference System (FIS) Tsukamoto

Fuzzy Inference System adalah suatu sistem yang menggunakan teori *fuzzy set* untuk memetakan *input* dan *output*. FIS mengimplementasikan sebuah set untuk *rules* yang mengklasifikasikan *input* kedalam *output* yang sesuai (Sihananto & Bachtiar, 2017). Penggambaran proses FIS Tsukamoto secara umum dapat dilihat pada Gambar 2.1.



Gambar 2.1 Arsitektur Fuzzy Logic Controller (Rebizant, et al., 2011)

Struktur dari metode FIS Tsukamoto antara lain:

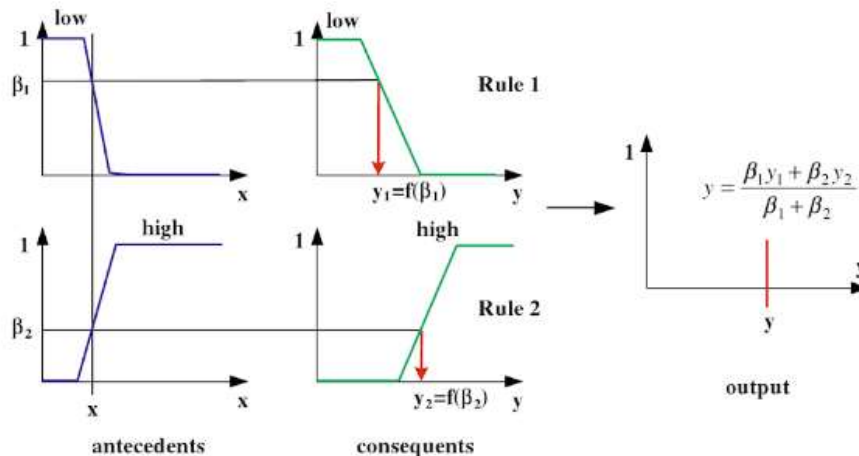
- Basis Pengetahuan: kumpulan *rule*-dalam bentuk pernyataan *IF ... THEN* yang dibuat oleh pakar dibidangnya.
- Fuzzifikasi: adalah proses untuk mengubah *input* sistem yang mempunyai nilai tegas menjadi variabel linguistik menggunakan fungsi keanggotaan yang disimpan dalam basis pengetahuan.
- Mesin inferensi: proses untuk mengubah *input fuzzy* menjadi *output fuzzy* dengan cara mengikuti aturan-aturan (*IF-THEN Rules*) yang telah ditetapkan pada basis pengetahuan *fuzzy*.
- Defuzzifikasi: mengubah *output fuzzy* yang diperoleh dari mesin inferensi menjadi nilai tegas menggunakan fungsi keanggotaan yang sesuai dengan saat dilakukan fuzzifikasi.

Secara umum bentuk model FIS tsukamoto adalah *IF ... AND ... THEN ...*. Dimana A, B, dan C adalah himpunan *fuzzy*. Misalkan diketahui 2 rule berikut:

If (x is A1) and (y is B1) then (z is C1)

If (x is A2) and (y is B2) then (z is C2)

Maka, skema penalarannya dapat dilihat pada Gambar 2.2.



Gambar 2.2 Skema penalaran FIS Tsukamoto (Rebizant, et al., 2011)

Penalaran FIS Tsukamoto dimulai dengan *fuzzifier* mengkonversikan set-set *input* kedalam fungsi keanggotaan *fuzzy* yang dapat membantu *knowledge base*, yang mana merupakan sebuah set pengetahuan yang didapatkan dari para ahli yang bisa digunakan untuk mengukur *output* di *Inference Engine* (Laribi, et al., 2004). *Output* yang didapatkan dari *inference engine* akan berbentuk nilai *fuzzy*. Agar *output* ini mudah dimengerti, *output* nya harus di defuzzifikasi menjadi nilai-nilai numerik oleh *defuzzifier* (Ren, et al., 2010). Kerangka komputasi pada FIS menggunakan aturan IF-THEN, dan penalaran *fuzzy*.

Metode FIS Tsukamoto mengaplikasikan penalaran monoton pada aturan-aturannya. Pada penalaran monoton, sistem hanya memiliki satu aturan, pada metode Tsukamoto, sistem terdiri atas beberapa aturan. Karena metode Tsukamoto menggunakan penalaran monoton, setiap konsekuen pada aturan yang berbentuk IF-THEN harus direpresentasikan dengan suatu himpunan *fuzzy* dengan fungsi keanggotaan yang monoton. *Output* hasil inferensi dari tiap aturan diberikan secara tegas (*crisp*) berdasarkan α -predikat (*fire strength*). Proses agregasi antar aturan dilakukan dan hasil akhirnya diperoleh dengan menggunakan defuzzifikasi dengan konsep rata-rata yang dibobotkan.

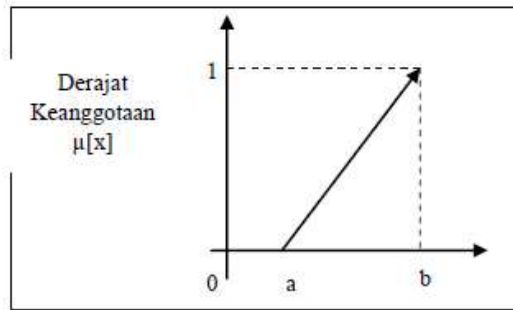
2.4.1.1 Fungsi Keanggotaan

Fungsi keanggotaan *fuzzy* adalah suatu kurva yang menunjukkan pemetaan titik-titik *input* data ke dalam derajat keanggotaannya yang nilainya bekisar antara 0 hingga 1. Beberapa fungsi keanggotaan *fuzzy* yaitu:

1. Representasi Linear

Representasi Linear adalah pemetaan *input* ke derajat keanggotaannya digambarkan sebagai suatu garis lurus. Pada representasi linear terdapat dua kemungkinan yaitu:

- Kenaikan himpunan dimulai pada nilai domain yang memiliki derajat keanggotaan nol (0) bergerak kearah kanan menuju nilai domain yang memiliki derajat keanggotaan yang lebih tinggi. Penggambaran kurva linear naik dapat dilihat pada Gambar 2.3.

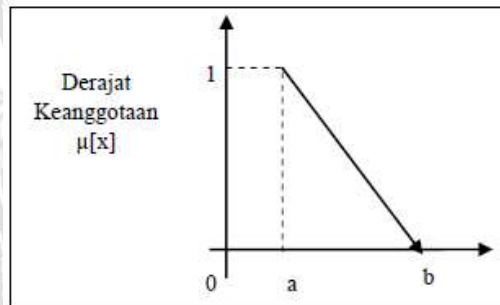


Gambar 2.3 Representasi Kurva Linear Naik

Fungsi keanggotaan:

$$\mu[x, a, b] = \begin{cases} 0, & x \leq a \\ \frac{x-a}{b-a}, & a \leq x \leq b \\ 1, & x \geq b \end{cases} \quad (2.1)$$

- b. Penurunan himpunan dimulai dari nilai domain dengan derajat keanggotaan tertinggi pada sisi kiri, kemudian bergerak menurun ke nilai domain yang memiliki derajat keanggotaan lebih rendah. Penggambaran kurva linear turun dapat dilihat pada Gambar 2.4.



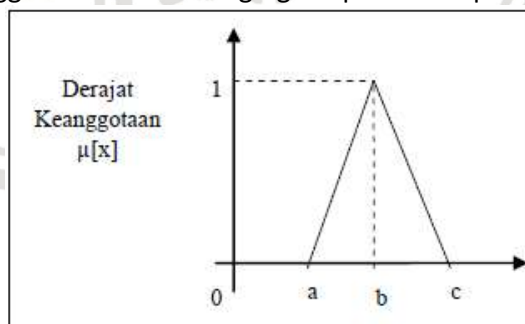
Gambar 2.4 Representasi Kurva Linear Turun

Fungsi Keanggotaan:

$$\mu[x, a, b] = \begin{cases} \frac{b-x}{b-a}, & a \leq x \leq b \\ 0, & x \geq b \end{cases} \quad (2.2)$$

2. Representasi Kurva Segitiga

Kurva Segitiga pada dasarnya terbentuk dari gabungan antara 2 garis (linear). Penggambaran kurva segitiga dapat dilihat pada Gambar 2.5.



Gambar 2.5 Representasi Kurva Segitiga

Fungsi Keanggotaan:

$$\mu[x, a, b] = \begin{cases} 0, & x \leq a \text{ atau } \geq c \\ \frac{x-a}{b-a}, & a \leq x \leq b \\ \frac{c-x}{c-b}, & b \leq x \leq c \end{cases} \quad (2.3)$$

2.4.1.2 Inferensi

Setelah fungsi keanggotaan untuk variabel masukan dan keluarannya ditentukan, basis aturan pengendalian dapat dikembangkan untuk menghubungkan aksi keluaran pengendali terhadap kondisi masukannya. Tahap ini disebut sebagai tahap inferensi, yakni bagian penentuan aturan dari sistem logika *fuzzy*. Sejumlah aturan dapat dibuat untuk menentukan aksi pengendali *fuzzy* (Wibawanto, 1998). Variabel masukan dan keluaran dapat dihubungkan oleh aturan IF-THEN yang terdapat pada basis aturan. Aturan IF-THEN ini akan direpresentasikan sebagai implikasi *fuzzy*. Dalam proses inferensi ini mencari nilai minimal (α -predikat) tiap nilai keanggotaan berdasarkan rule IF-THEN yang ada. Kemudian mencari nilai Z berdasarkan rule tersebut. Dalam kasus ini nilai Z adalah harga emas yang akan diramal. Perhitungan nilai Z dihitung menggunakan Persamaan 2.1.

$$Z = Z_{max} - \alpha \text{ predikat} \times (Z_{max} - Z_{min}) \quad (2.4)$$

2.4.1.3 Defuzzifikasi

Defuzzifikasi adalah sebuah proses yang mengubah aksi dari himpunan *fuzzy* menjadi satu nilai tunggal saja (Kusumadewi & Purnomo, 2004). Masukan dari proses defuzzifikasi adalah himpunan *fuzzy* yang didapatkan dari komposisi aturan *fuzzy*, sedangkan keluaran yang dihasilkan adalah suatu bilangan pada domain himpunan *fuzzy* tersebut. Sehingga jika diberikan sebuah himpunan *fuzzy* dalam rentang tertentu, maka harus dapat diambil suatu nilai tegas (*crisp*) tertentu sebagai keluarannya (Cristiono, 2005). Pada proses defuzzifikasi, nilai α -predikat akan dikalikan dengan nilai Z lalu di totalkan. Hasil perkalian ini kemudian dibagi dengan jumlah semua α -predikat untuk mendapatkan nilai Z final. Untuk persamaan defuzzifikasi dapat dilihat pada Persamaan 2.2.

$$Z = \frac{\sum \alpha\text{-predikat}_n \times Z_n}{\sum \alpha\text{-predikat}_n} \quad (2.5)$$

2.4.2 Algoritme Genetika

Algoritme Genetika adalah salah satu algoritme yang sering digunakan untuk menyelesaikan masalah yang cukup rumit. Algoritme ini seringkali digunakan untuk menyelesaikan masalah dalam berbagai bidang yang sering menghadapi masalah optimasi yang model matematikanya rumit. Algoritme Genetika sudah banyak diaplikasikan untuk permasalahan optimasi. Sebagai contoh Husada menggunakan Algoritme Genetika untuk mengoptimasi penjadwalan kuliah pengganti (Husada, et al., 2018)

Penggunaan Algoritme Genetika untuk menyelesaikan masalah yaitu dengan cara solusi dari suatu masalah tersebut harus dipetakan (*encoding*) menjadi string kromosom. String kromosom ini tersusun atas gen-gen yang menggambarkan beberapa variabel keputusan yang digunakan dalam solusi. Representasi

kromosom beserta fungsi *fitness* untuk mengetahui seberapa bagus sebuah kromosom menjadi layak untuk dimasukkan ke Algoritme Genetika.

Dengan mengaplikasikan proses genetika dan seleksi alam maka Algoritme Genetika akan menghasilkan kromosom terbaik setelah melewati beberapa generasi. Kromosom terbaik ini kemudian diuraikan (*decoding*) menjadi sebuah solusi yang mendekati optimum. Menurut Michalewicz (1996), dibandingkan dengan prosedur pencarian dan optimasi biasa, Algoritme Genetika memiliki perbedaan dalam beberapa hal, yaitu:

- Manipulasi dilakukan terhadap kode dari himpunan parameter (disebut kromosom), tidak secara langsung terhadap parameternya sendiri.
- Proses pencarian dilakukan dari beberapa titik dalam satu populasi, tidak dari satu titik saja.
- Proses pencarian menggunakan informasi dari fungsi tujuan.
- Pencariannya menggunakan *stochastic operator* yang sifatnya probabilistik, tidak menggunakan aturan yang bersifat deterministik.

Proses dalam Algoritme Genetika diawali dengan insialisasi, yaitu menciptakan individu-individu secara acak yang memiliki susunan gen (kromosom) tertentu. Kromosom ini mewakili solusi dari permasalahan yang akan diselesaikan. Reproduksi dilakukan untuk menghasilkan keturunan (*offspring*) dari individu-individu yang ada di populasi. Evaluasi digunakan untuk menghitung kebugaran (*fitness*) setiap kromosom. Semakin besar *fitness* maka kromosom akan semakin baik untuk dijadikan calon solusi. Seleksi dilakukan untuk memilih individu dari himpunan populasi dan *offspring* yang dipertahankan hidup pada generasi berikutnya. Fungsi probabilitas digunakan untuk memilih individu yang dipertahankan hidup. Individu yang lebih baik (mempunyai nilai *fitness* lebih besar) mempunyai peluang lebih besar untuk dipilih (Gen & Cheng, 1997).

2.4.3 Siklus Algoritme Genetika

Pada Algoritme Genetika terdapat 4 siklus utama yaitu inisialisasi, reproduksi, evaluasi, dan seleksi. Proses dalam Algoritme Genetika diawali dengan insialisasi, yaitu menciptakan individu-individu secara acak yang memiliki susunan gen (kromosom) tertentu. Kromosom ini mewakili solusi dari permasalahan yang akan diselesaikan. Reproduksi dilakukan untuk menghasilkan keturunan (*offspring*) dari individu-individu yang ada di populasi. Evaluasi digunakan untuk menghitung kebugaran (*fitness*) setiap kromosom. Semakin besar *fitness* maka kromosom akan semakin baik untuk dijadikan calon solusi. Seleksi dilakukan untuk memilih individu dari himpunan populasi dan *offspring* yang dipertahankan hidup pada generasi berikutnya. Fungsi probabilitas digunakan untuk memilih individu yang dipertahankan hidup. Individu yang lebih baik (mempunyai nilai *fitness* lebih besar) mempunyai peluang lebih besar untuk dipilih (Gen & Cheng, 1997).

Setelah melewati sekian generasi maka akan dihasilkan individu terbaik. Individu terbaik ini mempunyai susunan kromosom yang dapat dijadikan solusi yang terbaik. Maka dapat disimpulkan bahwa Algoritme Genetika menghasilkan suatu solusi optimal dengan melakukan pencarian diantara sejumlah alternatif titik optimum berdasarkan fungsi probabilitas (Michalewicz, 1996).

2.4.3.1 Inisialisasi

Inisialisasi dilakukan untuk membangkitkan himpunan solusi baru secara acak/*random* yang terdiri atas sejumlah *string* kromosom dan ditempatkan pada penampungan yang disebut populasi. Dalam tahap ini harus ditentukan ukuran populasi (*popSize*). Nilai ini menyatakan banyaknya individu/kromosom yang ditampung dalam populasi. Panjang setiap string kromosom dihitung berdasarkan presisi variabel yang solusi yang kita cari. Populasi awal sebagai daerah pencarian solusi optimal dibangkitkan secara acak. Representasi kromosom diperlukan untuk menjelaskan setiap individu dalam populasi. Setiap individu atau kromosom tersusun atas urutan gen dari suatu alfabet. Suatu alfabet dapat terdiri dari digit biner (0 dan 1), floating point, integer, simbol-simbol (seperti A, B, C), matriks, dan lain sebagainya.

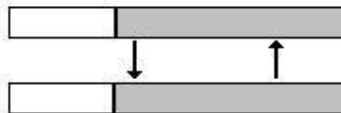
2.4.3.2 Reproduksi

Reproduksi dilakukan untuk menghasilkan keturunan dari individu-individu yang ada di populasi. Himpunan keturunan ini ditempatkan dalam penampungan *offspring*. Dua operator genetika yang digunakan dalam proses ini adalah tukar silang (*crossover*) dan mutasi (*mutation*). Ada banyak sekali metode yang *crossover* dan *mutation* yang dikembangkan para ahli dan bersifat spesifik terhadap masalah dan representasi kromosom yang digunakan.

a. Crossover

Dalam tahap ini akan ditentukan tingkat *crossover rate* (*cr*). Nilai ini menyatakan rasio *offspring* yang dihasilkan dari proses *crossover* terhadap ukuran populasi sehingga akan menghasilkan *offspring* sebanyak $cr \times popSize$.

Dalam penelitian ini menggunakan teknik *crossover one cut point*. Contoh kerja operator ini adalah dengan menentukan *crossover point* (gen tertentu). Kromosom baru pertama berisi gen pertama sampai gen *crossover point* dari kromosom induk pertama ditambah dengan gen dari *crossover point* sampai gen terakhir dari kromosom induk kedua. Kromosom baru kedua berisi gen pertama sampai gen *crossover point* dari induk kedua ditambahkan dengan gen dari *crossover point* sampai gen dari kromosom induk pertama. Adapun metode *crossover* ini dapat diilustrasikan pada Gambar 2.6 berikut:



Gambar 2.6 Proses *crossover* dengan satu *crossover point*

Dari ilustrasi di atas maka contoh penerapan metode *One-Point Crossover* adalah sebagai berikut:

Parent 1: 1 2 3 4 5 | 6 7 8 9

Parent 2: 4 5 3 6 8 | 9 7 2 1

Setelah proses *crossover* turunan yang dapat dihasilkan adalah dari kedua *parent* diatas adalah:

Child 1: 1 2 3 4 5 | 9 7 2 1

Child 2: 4 5 3 6 8 | 6 7 8 9

b. Mutasi

Mutasi merupakan proses mengubah nilai dari satu atau beberapa gen dalam suatu kromosom. Mutasi ini berperan untuk menggantikan gen yang hilang dari populasi akibat proses seleksi yang memungkinkan munculnya kembali gen yang tidak muncul pada inisialisasi populasi (Kusumadewi, 2003). Mutasi diterapkan dengan probabilitas sangat kecil. Peluang mutasi mengendalikan banyaknya gen baru yang akan dimunculkan untuk dievaluasi. Jika peluang mutasi terlalu kecil, banyak gen yang mungkin tidak pernah dievaluasi. Tetapi bila peluang mutasi terlalu besar, maka akan terlalu banyak gangguan acak, sehingga anak akan kehilangan kemiripan dari induknya.

Pada proses mutasi, digunakan nilai *mutation rate* (mr). Nilai *mutation rate* (mr) ini menyatakan rasio *offspring* yang dihasilkan dari proses mutasi terhadap ukuran populasi sehingga akan menghasilkan *offspring* sebanyak $mr \times popSize$.

Proses mutasi penelitian ini menggunakan *random mutation*. Proses ini dilakukan dengan memilih satu induk secara acak dari populasi. Kemudian gen ini akan ditambah atau dikurangi dengan bilangan random yang lebih kecil. Misalkan domain variabel x_j adalah $[min_j, max_j]$ dan *offspring* yang dihasilkan adalah $C = [x'_1 \dots x'_n]$, maka nilai gen *offspring* dapat dibangkitkan sebagai berikut:

$$x'_i = x'_i + r(max_j - min_j) \quad (2.6)$$

Dimana r adalah angka random dalam rentang tertentu.

2.4.3.3 Evaluasi dan Perhitungan *Fitness*

Evaluasi digunakan untuk menghitung kebugaran (*fitness*) setiap kromosom. Semakin besar *fitness* maka semakin baik kromosom tersebut dijadikan calon solusi. Proses evaluasi adalah sebuah proses untuk menghitung kebugaran (*fitness*) setiap kromosom. Kromosom yang dievaluasi merupakan kromosom gabungan dari populasi awal yang diinisialisasi dan hasil *offspring* yang didapatkan dari proses *crossover* dan mutasi. Setiap kromosom dievaluasi berdasarkan nilai *fitness* nya, semakin besar nilai *fitness* sebuah kromosom maka akan semakin baik pula solusi yang diberikan. Dari hasil proses evaluasi inilah nantinya akan digunakan pada proses seleksi.

Perhitungan *fitness* ini didapatkan dengan menghitung peramalan tiap fungsi keanggotaan, kemudian akan dicari nilai RMSE setiap individu yang telah dibahas sebelumnya, dan nilai RMSE inilah yang digunakan untuk mendapatkan nilai *fitness* setiap individu. Dalam penelitian ini, Nilai konstanta yang digunakan adalah 100. Adapun menghitung nilai *fitness* tersebut dirumuskan pada Persamaan 2.4.

$$fitness = \frac{100}{RMSE} \quad (2.7)$$

2.4.3.4 Seleksi

Seleksi dilakukan untuk memilih individu dari himpunan populasi dan *offspring* yang dipertahankan hidup pada generasi berikutnya. Semakin besar nilai

fitness sebuah kromosom maka semakin besar peluangnya untuk terpilih. Hal ini dilakukan agar terbentuk generasi berikutnya yang lebih baik. Metode seleksi yang sering digunakan adalah *roulette wheel*, *binary tournament*, dan *elitism*.

Seleksi yang digunakan pada penelitian ini adalah *elitism selection*. Secara umum, *elitism selection* akan mengumpulkan semua individu induk dan *offspring* dalam satu tempat. Individu terbaik sejumlah *popSize* kemudian akan lolos ke generasi selanjutnya. *Pseudo-code* dari *elitism selection* adalah sebagai berikut.

Pseudo-Code Elitism Selection	
1	PROCEDURE ElitismSelection
2	Input:
3	POP: himpunan individu pada populasi
4	pop_size: ukuran populasi
5	OS: himpunan individu hasil reproduksi (offspring)
6	menggunakan crossover and mutasi
7	
8	Output:
9	POP: himpunan individu pada populasi setelah proses
10	seleksi selesai
11	
12	/* menggabungkan individu pada POP dan OS ke dalam TEMP */
13	TEMP ← Merge (POP,OS)
14	/* mengurutkan fitness tiap individu secara ascending */
15	OrderAscending (Temp)
16	/* copy pop_size individu terbaik ke POP */
17	POP ← CopyBest (Temp, pop_size)
18	
19	END PROCEDURE

Kode Program 2.1 Pseudo-code Elitism Selection

2.5 Evaluasi Sistem

Evaluasi dilakukan untuk mengukur dan menguji seberapa jauh sistem peramalan yang dibuat mampu memprediksi data dengan cara mengevaluasi kesesuaian model peramalan terhadap suatu dataset yang diberikan. Sistem yang baik adalah sistem yang dapat memberikan hasil peramalan dengan nilai error sekecil mungkin.

Terdapat dua evaluasi yang digunakan dalam penelitian ini yang pertama adalah evaluasi untuk Algoritme Genetika. Evaluasi ini digunakan untuk menghitung kebugaran (*fitness*) setiap kromosom. Semakin besar *fitness* maka semakin baik kromosom tersebut dijadikan calon solusi.

Yang kedua adalah evaluasi untuk peramalan dengan metode Tsukamoto. Nilai evaluasi ini diukur dalam bentuk tingkat kesalahan (*error rate*) yang mana pada penelitian ini menggunakan *Root Mean Square Error* (RMSE) karena memberikan hasil yang relatif terhadap nilai yang sebenarnya sehingga nilai kesalahan yang didapatkan menjadi lebih stabil. Persamaan RMSE dirumuskan dalam Persamaan 2.5.

$$RMSE = \sqrt{\frac{\sum (Nilai \text{ Aktual} - Nilai Z)^2}{Juml \quad Data}} \quad (2.8)$$

BAB 3 METODOLOGI

Pada bab metodologi ini menjelaskan bagaimana tahapan-tahapan yang digunakan untuk mencapai tujuan penelitian. Pada bab ini juga akan menjelaskan tentang teknik pengumpulan data, perancangan sistem, implementasi sistem, dan pengujian sistem.

3.1 Tipe Penelitian

Tipe penelitian Optimasi Fungsi Keanggotaan FIS Tsukamoto dengan Algoritme Genetika pada Peramalan Harga Emas untuk *Stock Trading* termasuk dalam *non implementatif* karena penelitian ini bertujuan untuk menghasilkan suatu deskripsi.

Pendekatan dari penelitian ini berupa pendekatan pada deskripsi dari kegiatan penelitian non-implementatif. Untuk mencari hubungan antara fenomena dengan objek penelitian yang akan diteliti. Cara pendekatan penelitian ini bertujuan untuk menjelaskan suatu hubungan antara setiap komponen-komponen dalam penelitian ini. Setelah itu menghasilkan hasil dan analisis dari penelitian yang akan diteliti.

3.2 Strategi Penelitian

Strategi penelitian ini bertujuan untuk mengoptimasi fungsi keanggotaan fuzzy Tsukamoto dan meramalkan harga emas. Strategi penelitian ini dikontrol oleh peneliti dan pengembang yang bertanggung jawab. Metode yang digunakan adalah metode Algoritme Genetika untuk mengoptimasi fungsi keanggotaan dan FIS Tsukamoto untuk meramalkan harga emas.

3.3 Teknik Pengumpulan Data

Dalam penelitian ini, data didapatkan dari Forex Trading, Quandl, World Gold Council, dan US Energy Information Administration. Data yang terkumpul didapatkan dari tanggal 1 Januari 2013 sampai dengan 21 Juli 2016 dengan parameter yaitu kurs USD/GBP, CHF/USD, JPY/USD, EUR/USD dan harga emas. Untuk data latih menggunakan data hari Senin s/d Jumat dari tanggal 1 Januari 2013 s/d 11 Maret 2016. Sedangkan untuk data uji menggunakan data hari Senin s/d Jumat dari tanggal 13 Maret 2016 s/d 21 Juli 2016.

Data harga emas yang didapatkan dari Forex terkadang hilang atau kosong pada tanggal-tanggal tertentu. Biasanya data yang hilang ini terdapat pada hari libur internasional atau libur nasional negara-negara tertentu. Mengacu kepada hukum pasar, maka data yang hilang ini kemudian akan diisi dengan harga dari hari sebelumnya.

Semua data yang ada kemudian dilakukan *pre-processing* yaitu dengan cara mengurangi kurs mata uang pada hari tersebut dengan 3 hari sebelumnya dan harga emas akan dikurangi dengan harga sehari sebelumnya. Hal ini dilakukan untuk mengetahui pergerakan harga kurs mata uang berdasarkan 3 hari sebelumnya dan harga emas sehari sebelumnya. Karena indeks harga yang akan digunakan pada penelitian ini adalah indeks pergerakan harga, bukan indeks harga asli pada hari tersebut.

3.4 Perancangan Sistem

Dalam penelitian ini sistem yang akan dibuat digunakan untuk mengoptimasi fungsi keanggotaan FIS Tsukamoto dengan menggunakan Algoritme Genetika pada peramalan harga emas untuk *Stock Trading*. Pada tahap perancangan sistem menjabarkan mengenai perancangan kinerja sistem secara keseluruhan, dimulai dengan *input* yang diproses dan menghasilkan *output*. Pada proses yang dilakukan antara lain mengoptimasi fungsi keanggotaan FIS Tsukamoto dengan menggunakan Algoritme Genetika. Fungsi keanggotaan hasil optimasi tersebut kemudian akan digunakan untuk meramalkan harga emas menggunakan metode FIS Tsukamoto. Terdapat 3 alur perancangan sistem yaitu *input*, proses, dan *output*.

1. *Input*

Input atau masukan dari sistem untuk optimasi yaitu jumlah populasi, nilai *cr* dan *mr*, dan jumlah iterasi. Sedangkan untuk peramalan harga emas data masukan yang digunakan adalah data kurs mata uang dan harga emas.

2. Proses

Proses perhitungan pada sistem ini menggunakan Algoritme Genetika untuk optimasi. Sedangkan peramalan sistem akan menggunakan *rules* yang sudah dibangkitkan kemudian menghitungnya dengan menggunakan metode Tsukamoto.

3. *Output*

Output atau keluaran dari sistem ini adalah hasil optimasi fungsi keanggotaan dengan *fitness* terbaik dan perkiraan harga emas yang akan datang.

3.5 Pengujian dan Analisis

Pada tahapan ini adalah pengujian dari implementasi yang telah dilakukan untuk mendapatkan solusi yang paling mendekati optimal. Pada permasalahan optimasi fungsi keanggotaan, pengujian yang akan dilakukan adalah pengujian untuk populasi, pengujian kombinasi *cr* dan *mr*, dan pengujian iterasi. Pengujian dimulai dengan pengujian ukuran populasi yang terbaik, setelah nilai populasi yang terbaik didapatkan, dilanjutkan dengan pengujian kombinasi *cr* dan *mr*. Pada pengujian ini, populasi terbaik akan digunakan sebagai parameter untuk mendapatkan kombinasi *cr* dan *mr* yang terbaik. Pada pengujian konvergensi, nilai populasi terbaik serta kombinasi *cr* dan *mr* terbaik yang telah didapatkan sebelumnya dijadikan sebagai parameter pengujian. Kemudian akan dilakukan lagi pengujian sebanyak 10 kali menggunakan parameter terbaik dari pengujian-pengujian sebelumnya.

Pengujian pada penelitian mengenai optimasi fungsi keanggotaan FIS Tsukamoto dengan menggunakan Algoritme Genetika pada peramalan harga emas untuk *Stock Trading* termasuk pengujian kuantitatif karena data yang digunakan berupa angka. Penelitian ini dilakukan agar mampu menunjukkan bahwa aplikasi dapat bekerja sesuai dengan harapan. Setelah dilakukan optimasi fungsi keanggotaan menggunakan Algoritme Genetika dan peramalan harga emas

dengan menggunakan metode Tsukamoto, kemudian akan menghasilkan fungsi keanggotaan yang mendekati optimal dan perkiraan harga emas, maka sistem dapat dikatakan sesuai dengan tujuan.

Untuk pengujian peramalan sistem dilakukan dengan menggunakan nilai *Root Mean Square Error* (RMSE). Proses ini mengurangi nilai aktual dengan nilai prediksi lalu dikuadratkan. Hasil kuadrat kemudian akan dibagi dengan jumlah data. Lalu diakarkan untuk mendapatkan nilai RMSE. Setelah pengujian optimasi dan peramalan dilakukan, selanjutnya akan dilakukan analisis. Hasil semua pengujian ini akan dianalisis, analisis ini dilakukan untuk menjelaskan bagaimana hasil dari setiap pengujian yang dilakukan.

3.6 Penarikan Kesimpulan dan Saran

Penarikan kesimpulan dilakukan setelah semua tahap telah dilakukan (perancangan, implementasi, dan pengujian sistem). Kesimpulan diambil berdasarkan hasil pengujian sistem dan analisis terhadap sistem dengan tujuan untuk menjawab rumusan masalah yang telah ditetapkan sebelumnya. Selain kesimpulan terdapat tahap akhir yakni penulisan saran agar dapat digunakan oleh pembaca atau peneliti selanjutnya sebagai acuan untuk memperbaiki kesalahan yang terjadi serta memberikan pertimbangan untuk pengembangan selanjutnya.



BAB 4 PERANCANGAN ALGORITME

4.1 Deskripsi Masalah

Emas berperan penting dalam ekonomi dunia dan memiliki kedudukan yang baik di dalam bidang keuangan dan makroekonomi (Pierdzioch, et al., 2014). Emas memiliki nilai moneter dan dicari oleh bank pusat untuk dijadikan sebagai devisa negara yang mana bisa digunakan untuk berbagai tujuan (Gupta, et al., 2014). Pada bidang keuangan masa kini, emas digunakan untuk menjaga nilai ketika inflasi dan mengamankan aset ketika krisis (Hassani, et al., 2015). Hal ini disebabkan karena harga emas yang selalu stabil. Pengambilan keputusan dalam setiap transaksi dapat menentukan jumlah keuntungan ataupun kerugian yang didapat ketika melakukan pembelian atau penjualan. Dalam dunia investasi, resiko kerugian yang tidak diharapkan dalam pengambilan keputusan memiliki kemungkinan yang besar untuk terjadi. Karena pada dasarnya tidak ada investasi yang tidak beresiko (Indriani & Harjahdi, 2013). Untuk meminimalisir resiko yang didapat maka investor dan pedagang saham memerlukan pengetahuan tentang peramalan harga emas untuk mengetahui kapan harga emas akan naik atau akan turun agar dapat memaksimalkan keuntungan yang didapat.

Solusi dari permasalahan tersebut adalah dengan menyediakan suatu sistem yang dapat memprediksi harga emas. Sistem ini nantinya dapat mengetahui apakah harga emas yang akan datang itu naik atau turun. Penelitian ini menggunakan metode FIS Tsukamoto sebagai acuan untuk memprediksi harga emas dan Algoritma Genetika untuk mengoptimalkan fungsi keanggotaan FIS Tsukamoto tersebut.

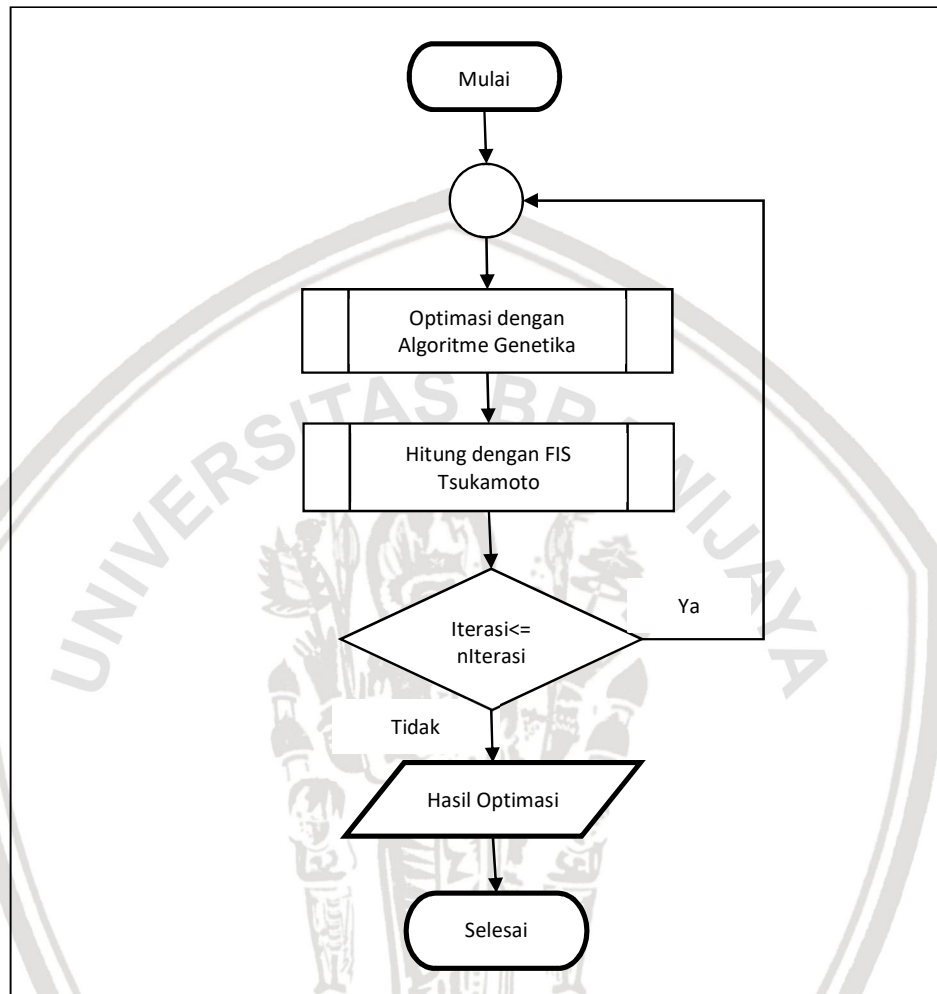
4.2 Deskripsi Umum Sistem

Optimasi fungsi keanggotaan *Fuzzy Tsukamoto* dengan menggunakan Algoritme Genetika pada peramalan harga emas untuk *Stock Trading* merupakan suatu sistem yang berbasis aplikasi. Sistem akan menampilkan tabel yang berisi informasi tentang kurs USD/GBP, CHF/USD, JPY/USD, EUR/USD dan GOLD. Kurs inilah yang nanti akan digunakan untuk meramal harga emas. Proses untuk meramal harga emas dilakukan melalui beberapa proses yaitu mencari selisih kurs mata uang dari 3 hari sebelumnya, menentukan fungsi keanggotaan, mencari setiap nilai keanggotaan, melakukan perhitungan berdasarkan rule yang sudah ada. Setelah hasil peramalan didapatkan, akan dilakukan proses optimasi untuk fungsi keanggotaannya dengan menggunakan metode Algoritme Genetika guna mendapatkan fungsi keanggotaan yang terbaik.

4.3 Diagram Alir Sistem

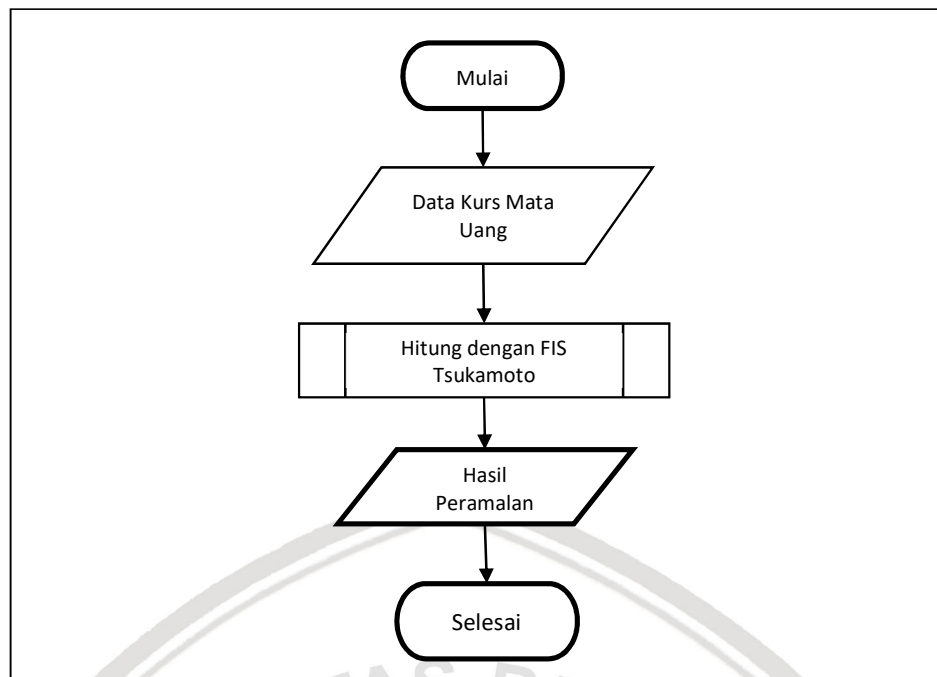
Pada sub-bab ini akan ditampilkan seluruh diagram alir yang digunakan pada sistem. Metode dalam sistem yang dikembangkan ini adalah kombinasi antara FIS Tsukamoto dan Algoritme Genetika. Algoritme Genetika digunakan untuk mendapatkan fungsi keanggotaan yang akan digunakan untuk peramalan dengan metode Tsukamoto. Terdapat 4 kurs mata uang dan 1 harga emas yang digunakan sebagai data. Fungsi keanggotaan tiap kurs dan harga emas ini kemudian

direpresentasikan sebagai individu dalam Algoritme Genetika. Sistem kemudian akan menghitung data dengan fungsi keanggotaan yang direpresentasikan tersebut dengan metode FIS Tsukamoto. Hasil dari metode Tsukamoto ini kemudian digunakan untuk mendapatkan nilai *fitness*. Proses ini dilakukan berulang-ulang sampai iterasi yang ditentukan. Diagram alir sistem optimasi fungsi keanggotaan dimodelkan dengan diagram alir pada Gambar 4.1.



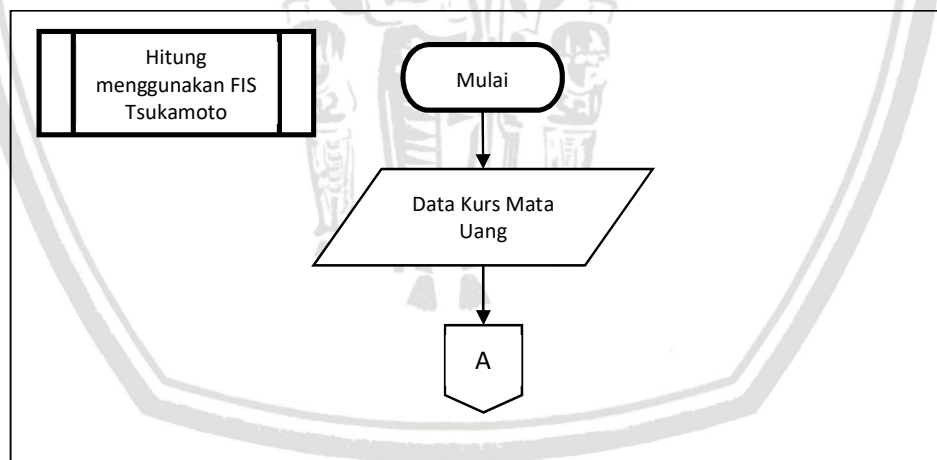
Gambar 4.1 Diagram Alir Sistem Optimasi Secara Umum

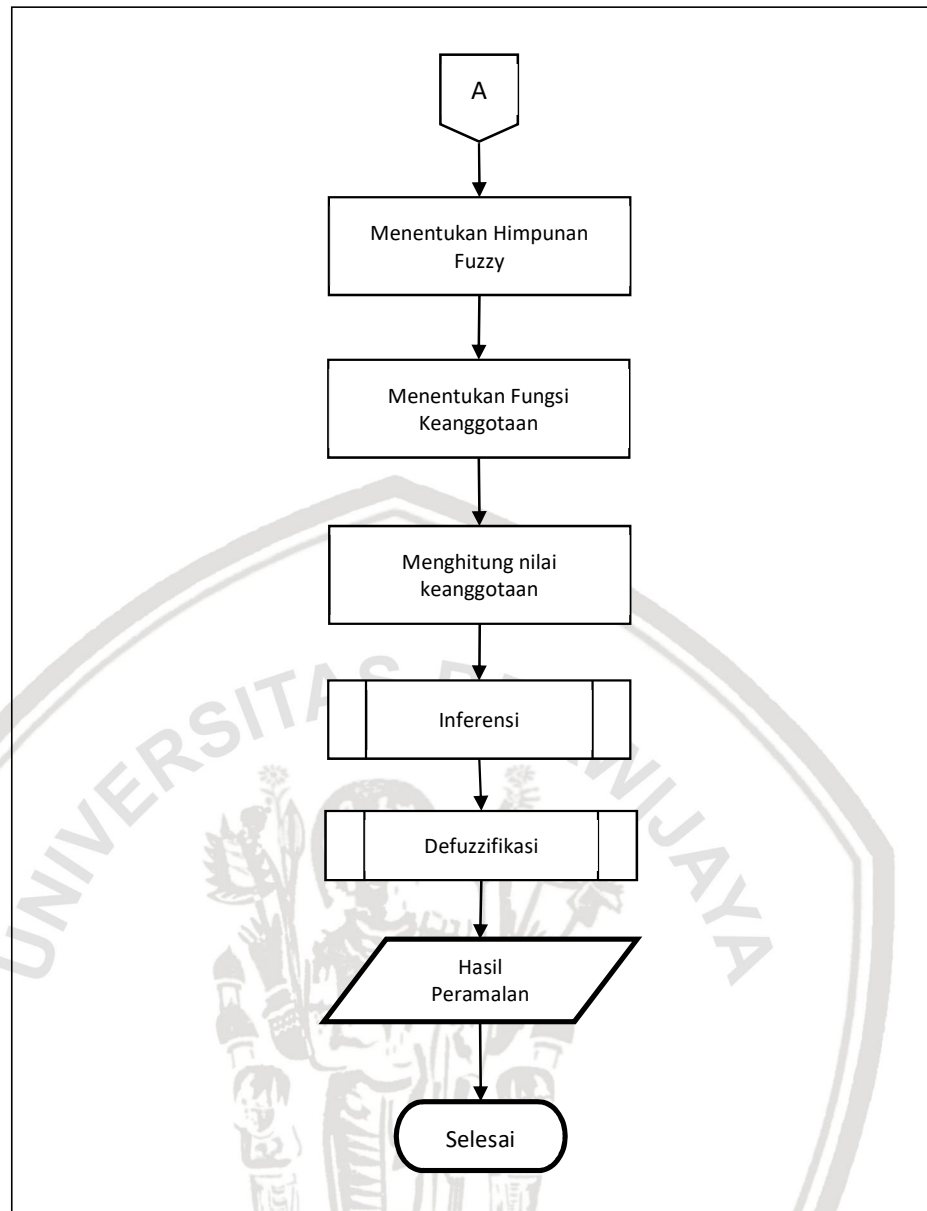
Untuk peramalan harga emas, sistem yang dikembangkan adalah FIS Tsukamoto. Pada metode Tsukamoto, data kurs mata uang dihitung selisihnya dengan harga 3 hari sebelumnya. Kemudian menghitung nilai keanggotaan berdasarkan fungsi keanggotaan yang sudah ditentukan sebelumnya. Setelah itu dilakukan inferensi untuk mencari nilai α -predikat dan nilai $Z(n)$ berdasarkan rule IF-THEN yang sudah ada. Setelah itu dilakukan proses Defuzzifikasi untuk mencari nilai Z final yang merupakan harga peramalan. Diagram alir FIS Tsukamoto dimodelkan pada Gambar 4.2.



Gambar 4.2 Diagram Alir Sistem Peramalan Secara Umum

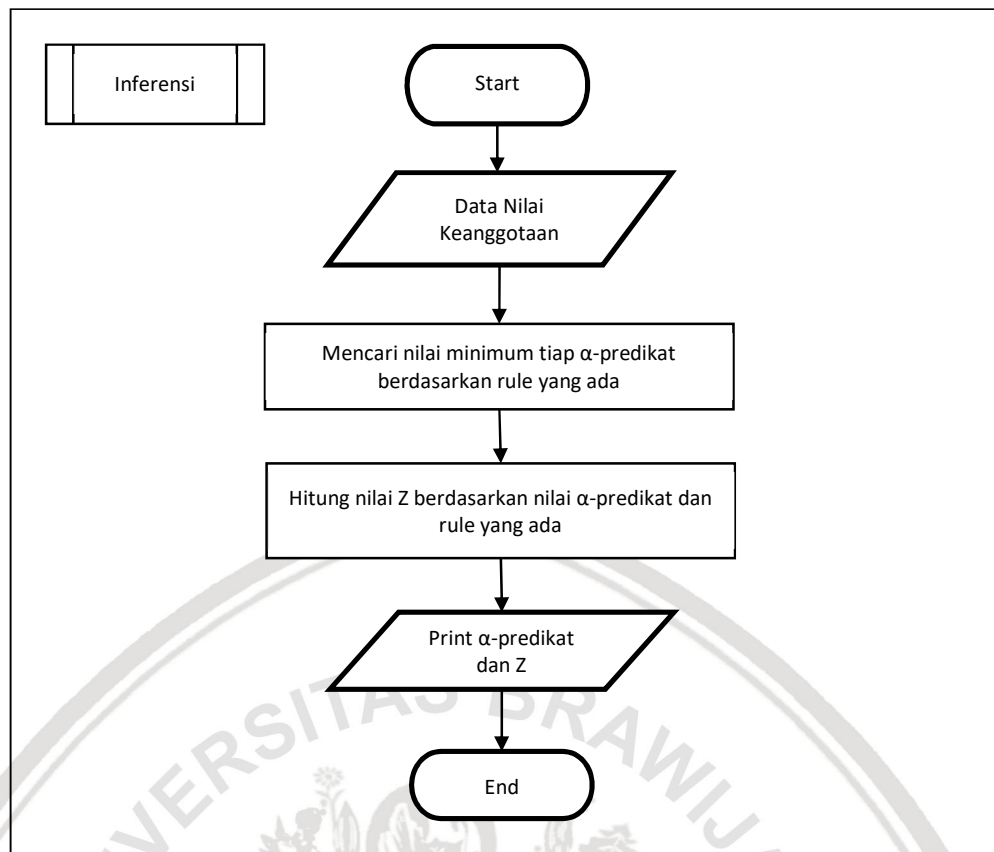
Metode dalam sistem yang dikembangkan ini adalah kombinasi antara FIS Tsukamoto dan Algoritme Genetika. Pada metode Tsukamoto, data kurs mata uang dihitung selisihnya dengan harga 3 hari sebelumnya. Kemudian menghitung nilai keanggotaan berdasarkan fungsi keanggotaan yang sudah ditentukan sebelumnya. Setelah itu dilakukan inferensi untuk mencari nilai α -predikat dan nilai $Z(n)$ berdasarkan rule IF-THEN yang sudah ada. Setelah itu dilakukan proses Defuzzifikasi untuk mencari nilai Z final yang merupakan harga peramalan Diagram alir FIS Tsukamoto dimodelkan pada Gambar 4.3.





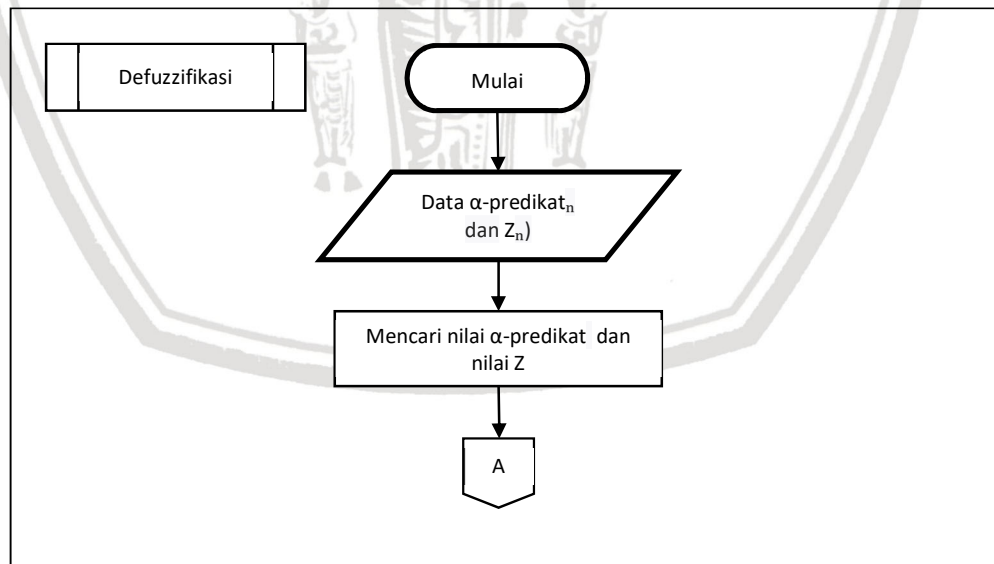
Gambar 4.3 Diagram Alir Metode FIS Tsukamoto

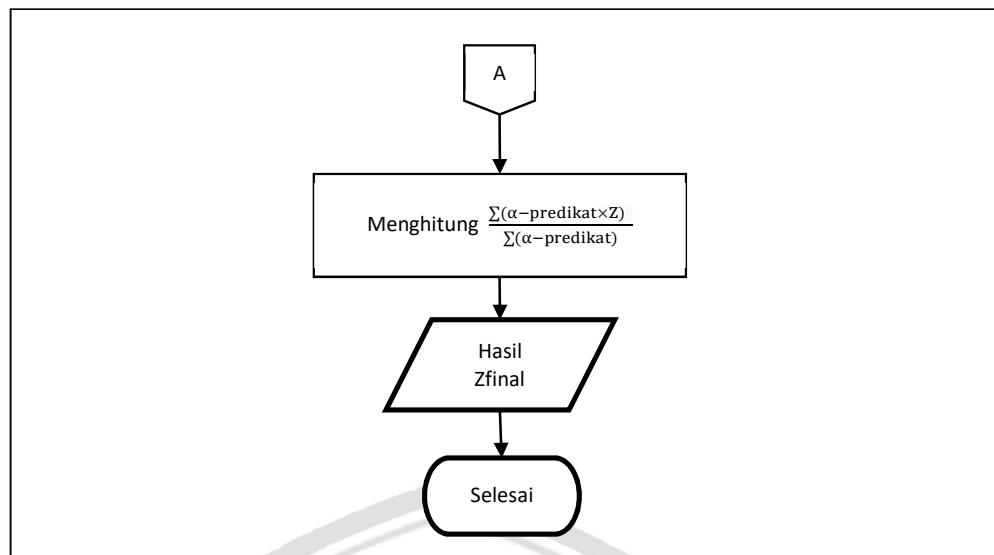
Setelah nilai keanggotaan didapatkan selanjutnya adalah mencari inferensi. Dalam proses inferensi ini mencari nilai minimal (α -predikat) tiap nilai keanggotaan berdasarkan rule IF-THEN yang ada. Kemudian mencari nilai Z berdasarkan rule tersebut. Dimana nilai Z ini adalah indeks harga emas yang akan diramalkan. Proses Inferensi dapat dilihat dalam Gambar 4.4.



Gambar 4.4 Diagram Alir Inferensi

Setelah melakukan inferensi selanjutnya adalah melakukan proses defuzzifikasi. Dalam proses ini nilai α -predikat dan nilai Z dikalikan. Lalu hasil kali tersebut dijumlahkan. Hasil kali ini kemudian dibagi dengan jumlah semua α -predikat untuk mendapatkan nilai Z final. Proses defuzzifikasi dapat dilihat pada Gambar 4.5.

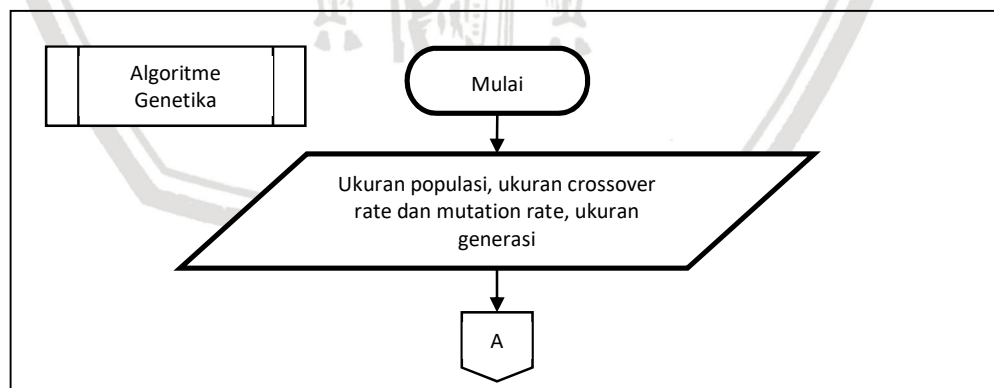


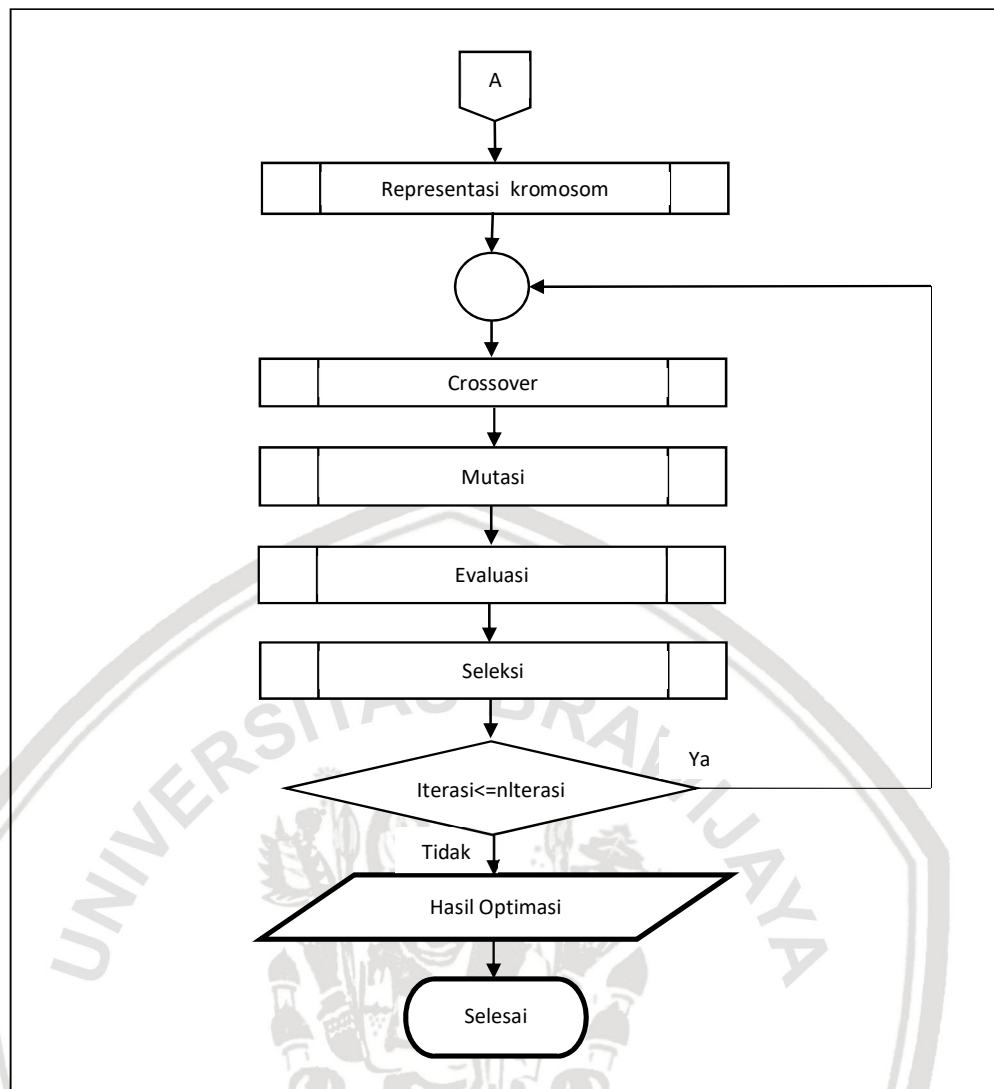


Gambar 4.5 Diagram Alir Proses Defuzzifikasi

Setelah mendapatkan hasil dari FIS selanjutnya akan dioptimasi dengan Algoritme Genetika. Setiap fungsi keanggotaan akan dijadikan kromosom, kemudian dibangkitkan sebanyak populasi untuk selanjutnya dihitung dengan Algoritme Genetika. Langkah pertama penyelesaian dengan Algoritme Genetika adalah melakukan insialisasi populasi awal secara random sebanyak n individu. Kemudian pada langkah kedua, melakukan reproduksi dengan crossover dengan memilih dua induk (parent) secara random dari populasi yang ada. Selanjutnya reproduksi dilakukan dengan cara mutasi pada kromosom induk yang dipilih secara acak.

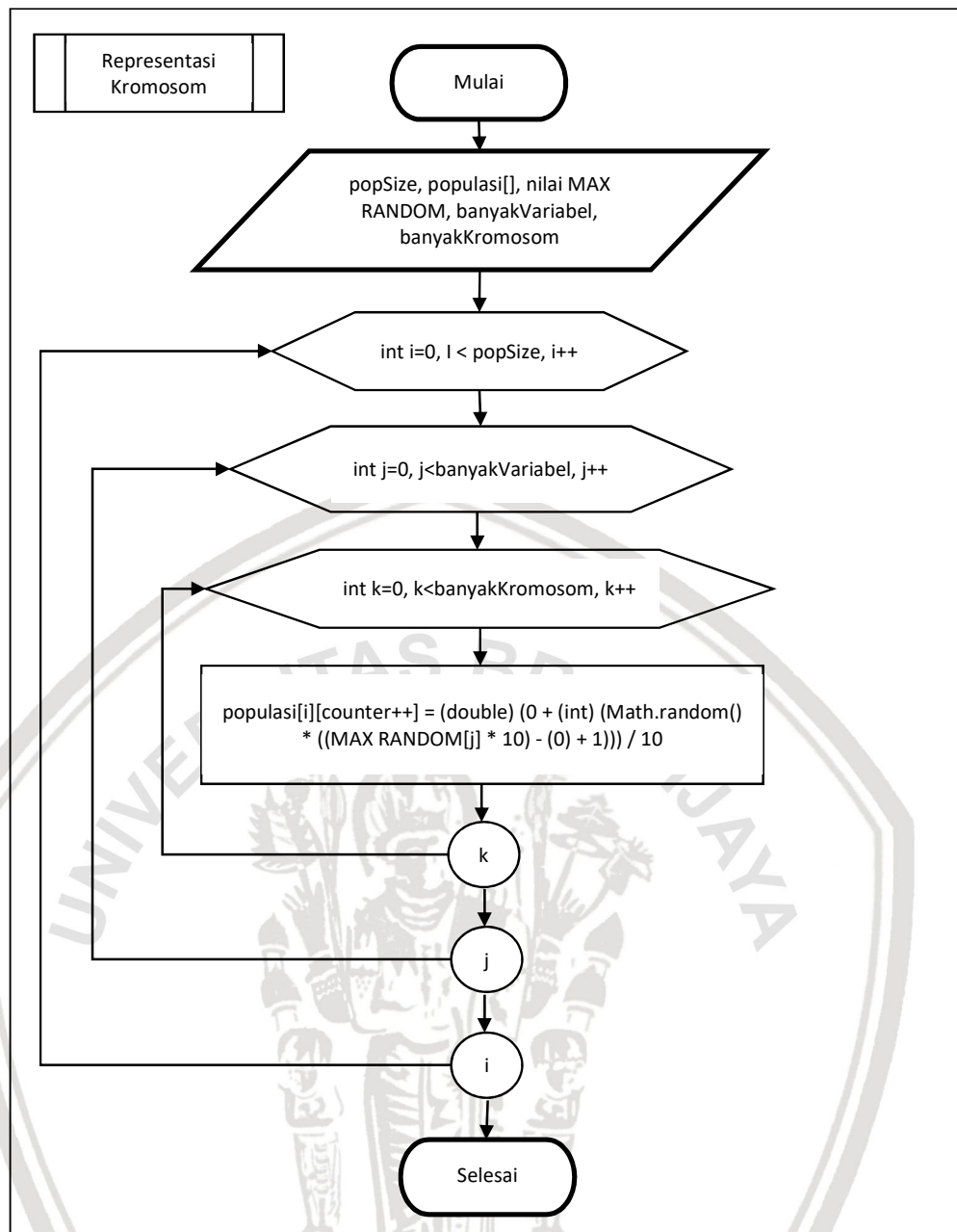
Langkah selanjutnya adalah mencari nilai *fitness* pada masing-masing kromosom. Pada langkah terakhir, melakukan proses seleksi dengan metode *elitism selection* dengan tujuan untuk menentukan individu yang akan dibangkitkan pada generasi berikutnya. Hasil akhir pada Algoritme Genetika ini dengan mendapatkan kromosom yang terbaik pada generasi terakhir. Apabila kondisi pada akhir terpenuhi, maka hasil yang didapat merupakan solusi yang terbaik pada seluruh generasi. Diagram alir Algoritme Genetika dapat dilihat pada Gambar 4.6.





Gambar 4.6 Diagram Alir Metode Algoritma Genetika

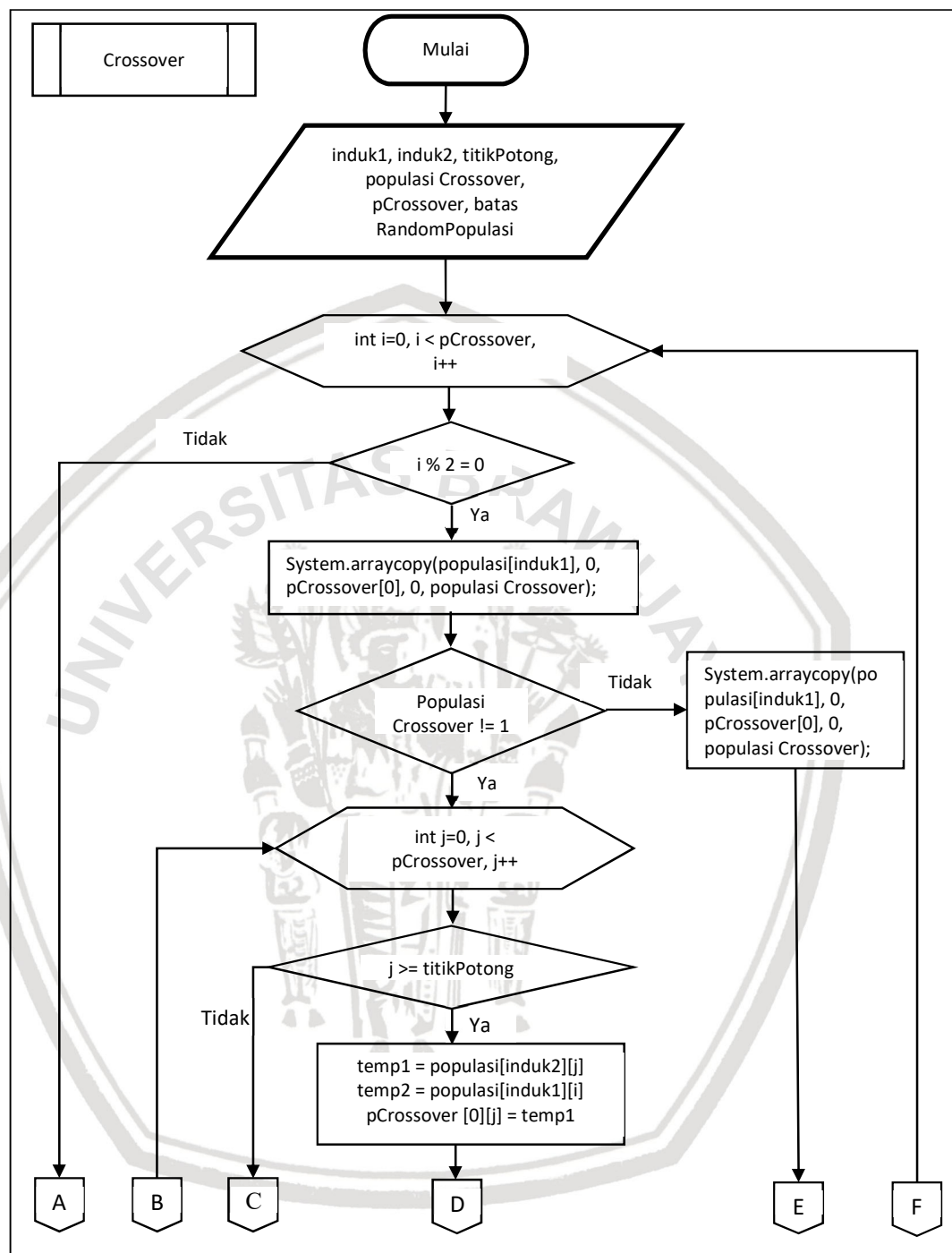
Langkah awal pada proses Algoritma Genetika adalah representasi kromosom. Terdapat beberapa tahapan pada representasi kromosom ini. Sistem akan menerima masukan jumlah populasi yang akan digunakan dalam proses, lalu setiap gen dibangkitkan dengan nilai *random* dalam rentang yang telah ditentukan pada masing-masing variabel. Setelah itu, gen tersebut akan digabungkan untuk kemudian dibentuk menjadi sebuah kromosom atau individu. Kromosom yang dibangkitkan sesuai dengan jumlah populasi yang telah dimasukkan diawal, dan kemudian akan dilanjutkan ke proses selanjutnya. Diagram alir untuk proses representasi kromosom dapat dilihat pada Gambar 4.7.

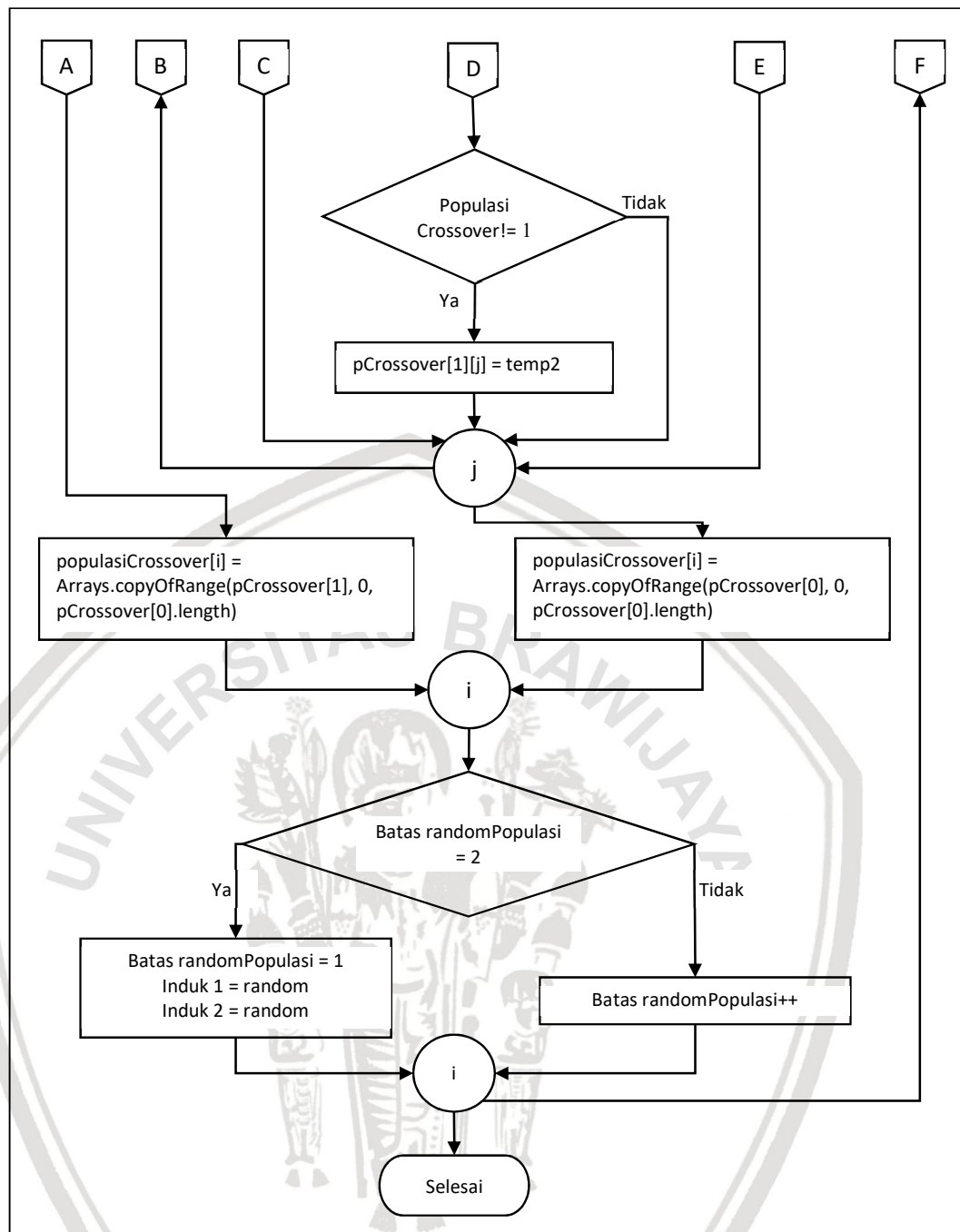


Gambar 4.7 Diagram Alir Inisialisasi Kromosom

Setelah kromosom ditentukan selanjutnya kromosom ini akan dikembangkan dengan proses *crossover*. Proses ini dimulai dengan mengambil 2 kromosom yang telah diinisialisasi pada populasi awal secara acak. Kromosom tersebut akan direproduksi dengan *one cut point*. Individu yang akan dikembangkan pada proses reproduksi *crossover* ini sebanyak $offspring = cr \times popsize$, sehingga proses *crossover* ini akan menghasilkan keturunan sebanyak nilai *offspring*. Kemudian, dua individu akan dipilih secara acak untuk dijadikan sebagai *parent*. Sistem akan memilih titik potong secara acak, sehingga kedua *parent* ini memiliki dua bagian yaitu bagian kiri dan kanan. Kemudian bagian kanan dari *parent* ini akan ditukar untuk menghasilkan *offspring*. Sehingga *offspring* yang

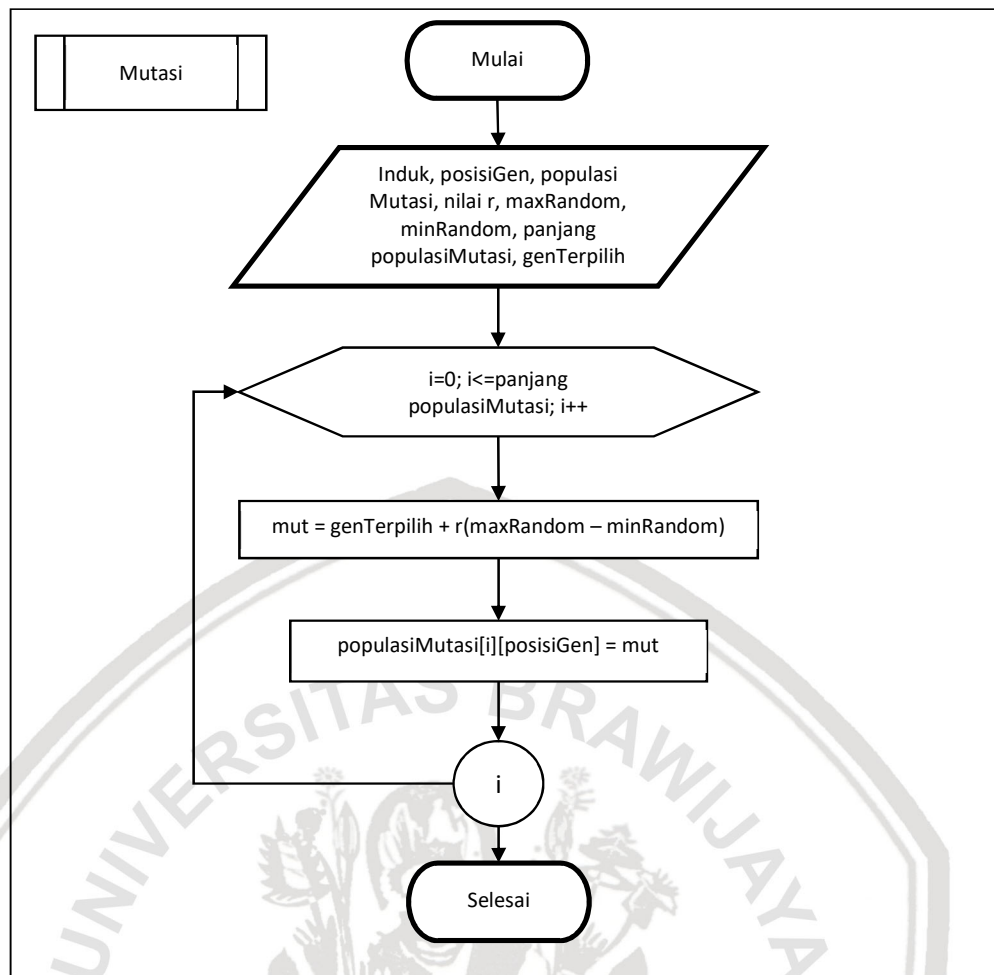
dihasilkan dari proses *crossover* ini adalah bagian kiri dari satu *parent* dan bagian kanan dari *parent* yang lain. *Offspring* yang dihasilkan adalah sejumlah nilai yang telah ditentukan berdasarkan nilai *cr* dikalikan dengan *popsiz*e. Diagram alir proses *crossover* dapat dilihat pada Gambar 4.8.





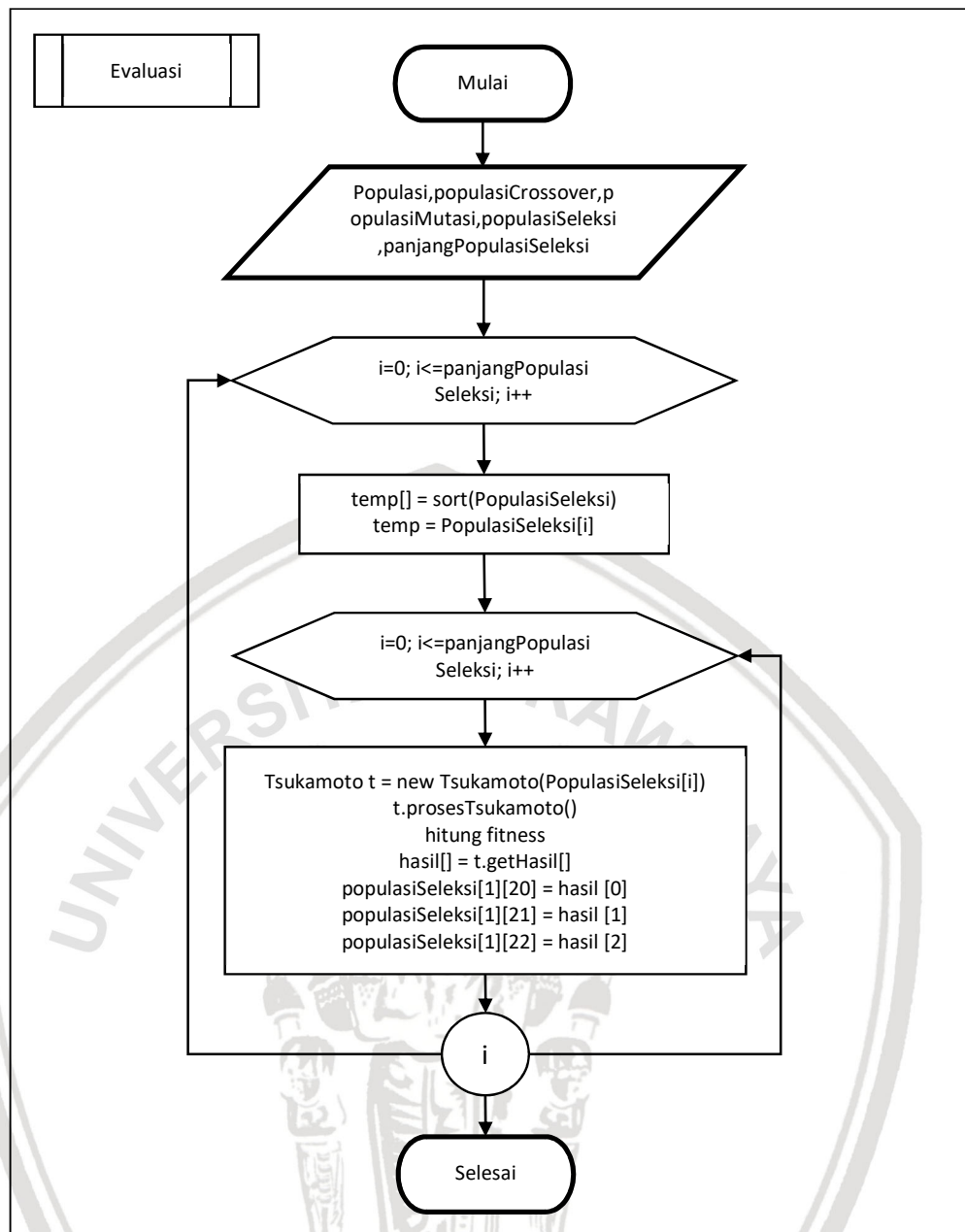
Gambar 4.8 Diagram Alir Crossover

Setelah dikembangkan dengan proses *crossover*, selanjutnya akan dikembangkan lagi dengan proses mutasi. Metode mutasi yang digunakan adalah *random mutation*. Individu yang akan dikembangkan sebanyak $offspring = mr \times popsize$. Kemudian akan dipilih satu induk secara acak dan di induk ini akan dipilih satu gen yang akan dimutasi. Gen yang terpilih ini akan ditambah atau dikurangi dengan bilangan random yang kecil. Lalu akan ditentukan nilai maksimum dan minimum dari variabel tersebut dan akan dihitung menggunakan Persamaan 2.3. Diagram alir proses mutasi ini dapat dilihat pada Gambar 4.9.



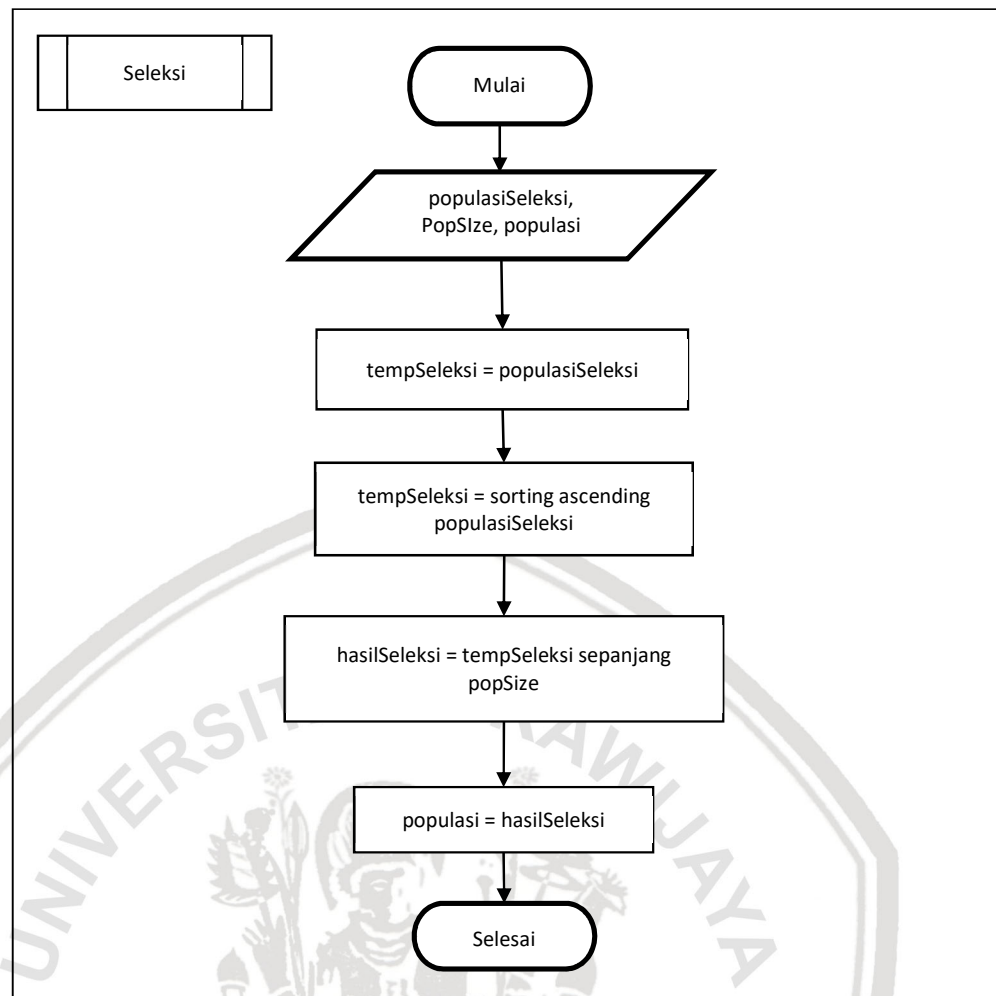
Gambar 4.9 Proses Mutasi

Setelah dikembangkan lagi dengan mutasi, kemudian akan dilakukan proses evaluasi. Proses evaluasi adalah sebuah proses untuk menghitung kebugaran (*fitness*) setiap kromosom. Kromosom yang dievaluasi merupakan kromosom gabungan dari populasi awal yang diinisialisasi dan hasil *offspring* yang didapatkan dari proses *crossover* dan mutasi. Setiap kromosom dievaluasi berdasarkan nilai *fitness* nya, semakin besar nilai *fitness* sebuah kromosom maka akan semakin baik pula solusi yang diberikan. Dari hasil proses evaluasi inilah nantinya akan digunakan pada proses seleksi. Perhitungan *fitness* ini didapatkan dengan menghitung peramalan tiap fungsi keanggotaan, kemudian akan dicari nilai RMSE setiap individu menggunakan Persamaan 2.5, dan nilai RMSE inilah yang digunakan untuk mendapatkan nilai *fitness* menggunakan persamaan 2.4. Diagram alir proses evaluasi dapat dilihat pada Gambar 4.10.



Gambar 4.10 Diagram Alir Evaluasi

Proses seleksi adalah sebuah proses untuk memilih individu dari himpunan populasi dan *offspring* yang akan digunakan untuk iterasi selanjutnya. Semakin tinggi nilai *fitness* sebuah kromosom maka peluangnya untuk terpilih akan semakin besar. Hal ini bertujuan agar generasi yang terbentuk berikutnya lebih baik dari generasi sekarang. Proses seleksi penelitian ini menggunakan metode *elitism*, metode *elitism* adalah sebuah metode yang menyaring kromosom terbaik berdasarkan nilai *fitness* yang paling tinggi yang akan dipertahankan untuk iterasi selanjutnya. Diagram alir proses seleksi dapat dilihat pada Gambar 4.11



Gambar 4.11 Diagram Alir Seleksi

4.4 Manualisasi Perhitungan Data

Dalam contoh perhitungan data manualisasi ini terdapat 5 jenis data. Yaitu data kurs 4 mata uang USD/GBP, CHF/USD, JPY/USD berdasarkan 3 hari sebelumnya dan data harga emas berdasarkan 1 hari sebelumnya. Dataset yang digunakan sebanyak 1000 data latih dan 100 data uji. Tabel 4.1 menunjukkan 20 data latih dari keseluruhan data latih. Tabel 4.2 menunjukkan 20 data uji dari keseluruhan data uji. Untuk lebih lengkapnya tentang kedua data ini dapat dilihat pada Lampiran A.

Tabel 4.1 Data Latih

No	Tanggal	USD/GBP	CHF/USD	JPY/USD	EUR/USD	Gold
1	01/01/2013	8,96	10,80	-0,80	13,45	36,25
2	02/01/2013	-15,41	15,18	-0,37	18,71	-14,25
3	03/01/2013	-17,91	17,57	0,03	20,98	-31,50
4	04/01/2013	-18,46	-3,20	0,00	-4,16	0,00

No	Tanggal	USD/GBP	CHF/USD	JPY/USD	EUR/USD	Gold
5	06/01/2013	3,97	-4,26	0,34	-5,52	-2,75
6	07/01/2013	-0,62	-7,91	-0,67	-9,60	10,75
7	08/01/2013	-5,25	-8,81	0,59	-10,98	1,75
8	09/01/2013	4,09	-7,75	0,24	-9,62	17,25
9	10/01/2013	6,53	2,83	0,45	3,33	-17,50
10	11/01/2013	11,16	3,77	-0,92	4,87	0,00
11	13/01/2013	-8,58	1,00	-1,20	1,24	9,00
12	14/01/2013	-5,24	2,43	-1,48	3,05	14,00
13	15/01/2013	-12,44	0,71	-1,11	0,67	-4,25
14	16/01/2013	-10,02	3,47	-0,83	4,30	-1,25
15	17/01/2013	-21,92	-0,32	-0,13	-0,46	13,50
16	18/01/2013	-14,72	12,16	0,43	14,40	0,00
17	20/01/2013	-13,92	16,93	1,55	20,19	-1,00
18	21/01/2013	-1,63	17,35	1,17	20,79	3,00
19	22/01/2013	-2,26	2,30	0,82	2,78	-0,25
20	23/01/2013	-5,50	-2,47	-0,31	-3,01	-19,25

Tabel 4.2 Data Uji

No	Tanggal	USD/GBP	CHF/USD	JPY/USD	EUR/USD	Gold
1	13/03/2016	-0,57	12,71	0,94	10,56	-22
2	14/03/2016	-22,37	12,45	0,67	1,27	-10,75
3	15/03/2016	-12,48	5,77	0,32	-2,39	-3,5
4	16/03/2016	19,60	-4,01	-0,31	-11,45	38
5	17/03/2016	31,17	-5,46	-0,32	-3,92	-14,4
6	18/03/2016	21,28	-3,30	-0,36	-0,71	0
7	20/03/2016	-10,80	-6,66	-0,40	-1,30	-7,2
8	21/03/2016	-24,31	-18,51	-0,94	-24,81	7,6
9	22/03/2016	-35,11	-24,11	-1,04	-33,36	-34,9
10	23/03/2016	-24,00	-20,75	-1,00	-32,77	3,4
11	24/03/2016	-9,39	-9,35	-0,06	-1,50	0
12	25/03/2016	1,42	-8,70	0,56	1,67	0

No	Tanggal	USD/GBP	CHF/USD	JPY/USD	EUR/USD	Gold
13	27/03/2016	13,32	-15,43	0,30	-9,14	0
14	28/03/2016	25,53	-6,44	-0,03	-7,21	5
15	29/03/2016	23,59	0,74	-0,01	-1,11	10,25
16	30/03/2016	10,37	7,47	0,25	9,70	0,75
17	31/03/2016	-16,19	1,76	0,15	2,22	-23,4
18	01/04/2016	-10,21	-4,83	-0,41	-6,20	0
19	03/04/2016	-21,07	-6,76	-0,68	-15,79	6,15
20	04/04/2016	-8,65	-8,32	-0,60	-13,20	11,5

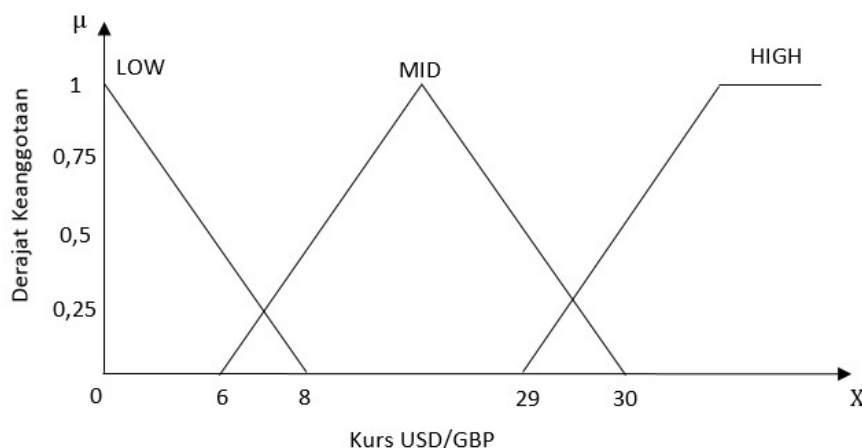
4.4.1 FIS Tsukamoto

Metode FIS Tsukamoto digunakan untuk meramalkan harga emas. Pertama akan ditentukan himpunan *fuzzy* dari masing-masing kurs mata uang. Setiap data kurs mata uang yang digunakan akan dihitung nilai keanggotaannya. Selanjutnya akan dilakukan inferensi sesuai dengan *rule* IF-THEN yang ada. Setelah itu akan dilakukan proses defuzzifikasi untuk mendapatkan indeks harga yang diramal.

4.4.1.1 Menentukan Himpunan Fuzzy

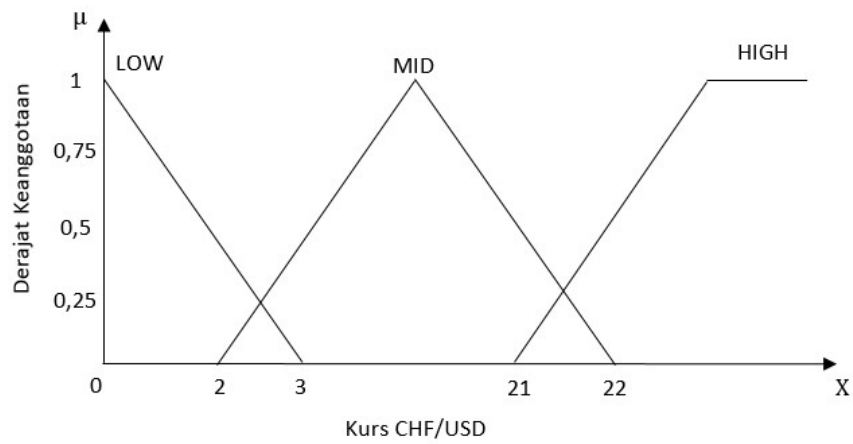
Dalam percobaan ini terdapat 5 himpunan *fuzzy* yaitu himpunan *fuzzy* USD/GBP, CHF/USD, JPY/USD, EUR/USD dan Gold. Setiap himpunan *fuzzy* bernilai positif apabila indeks naik (Up) dan akan bernilai negatif apabila indeks turun (Down). Himpunan *fuzzy* setiap variabel dapat dilihat pada Gambar 4.12 sampai Gambar 4.16. Pada Gambar 4.12 sampai Gambar 4.16 ini akan menggunakan fungsi keanggotaan berdasarkan nilai acak yang dibangkitkan.

- USD/GBP



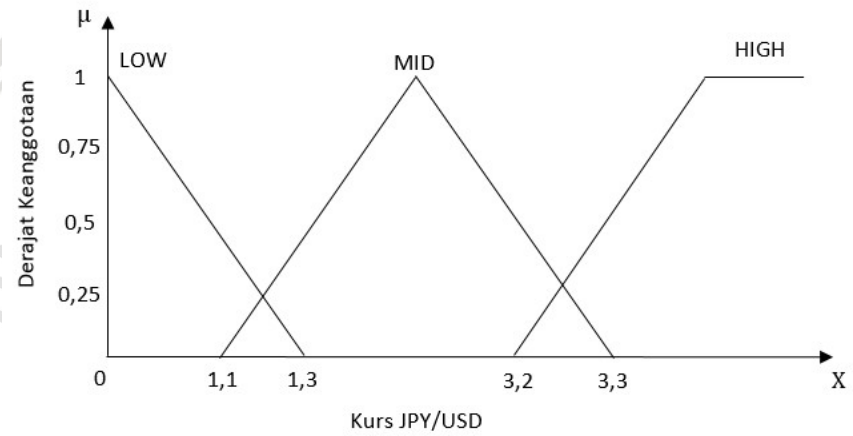
Gambar 4.12 Fungsi Keanggotaan USD/GBP

- CHF/USD



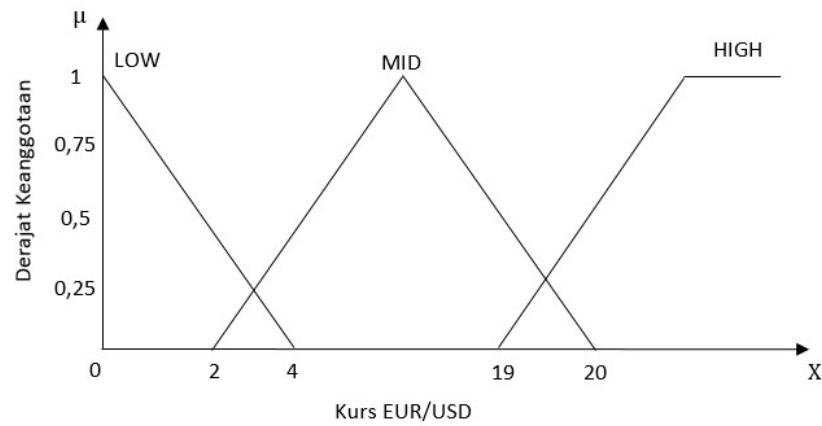
Gambar 4.13 Fungsi Keanggotaan CHF/USD

- JPY/USD



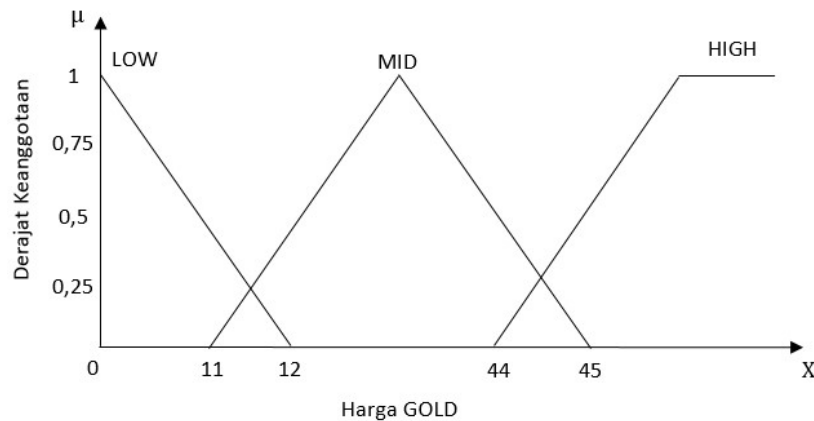
Gambar 4.14 Fungsi Keanggotaan JPY/USD

- EUR/USD



Gambar 4.15 Fungsi Keanggotaan EUR/USD

- GOLD



Gambar 4.16 Fungsi Keanggotaan GOLD

4.4.1.2 Menentukan Fungsi Keanggotaan

Dalam contoh perhitungan manual ini, fungsi keanggotaan tiap variabel dibagi menjadi 6 kategori yaitu Down Low, Down Mid, Down High, Up Low, Up Mid, dan Up High. Untuk perhitungan fungsi keanggotaan didapatkan dari himpunan *fuzzy* tiap variabel.

- Fungsi Keanggotaan USD/GBP:

$$\mu_{DownLow}[x] = \begin{cases} \frac{(-8) - x}{(-8) - 0}, & 0 \geq x \geq (-8) \\ 0, & x \leq (-8) \end{cases}$$

$$\mu_{DownMid}[x] = \begin{cases} 0, & x \geq -6 \text{ atau } x \leq -30 \\ \frac{x - (-6)}{(-17,5) - (-6)}, & -6 \geq x \geq (-17,5) \\ \frac{(-30) - x}{(-30) - (-17,5)}, & (-17,5) \geq x \geq (-30) \end{cases}$$

$$\mu_{DownHigh}[x] = \begin{cases} 0, & x \geq (-29) \\ \frac{x - (-29)}{(-39,50) - (-29)}, & (-29) \geq x \geq (-39,50) \\ 1, & x \leq (-40) \end{cases}$$

$$\mu_{UpLow}[x] = \begin{cases} \frac{8 - x}{8 - 0}, & 0 \leq x \leq 8 \\ 0, & x \geq 8 \end{cases}$$

$$\mu_{UpMid}[x] = \begin{cases} 0, & x \leq 6 \text{ atau } x \geq 30 \\ \frac{x - 6}{17,5 - 6}, & 6 \leq x \leq 17,5 \\ \frac{30 - x}{30 - 17,5}, & 17,5 \leq x \leq 30 \end{cases}$$

$$\mu_{UpHigh}[x] = \begin{cases} 0, & x \leq 29 \\ \frac{x - 29}{33,97 - 29}, & 29 \leq x \leq 33,97 \\ 1, & x \geq 33,97 \end{cases}$$

- Fungsi Keanggotaan CHF/USD

$$\mu_{DownLow}[x] = \begin{cases} \frac{(-3) - x}{(-3) - 0}, & 0 \geq x \geq (-3) \\ 0, & x \leq (-3) \end{cases}$$

$$\mu_{DownMid}[x] = \begin{cases} 0, & x \geq -1 \text{ atau } x \leq -22 \\ \frac{x - (-1)}{(-11,5) - (-1)}, & -1 \geq x \geq (-11,5) \\ \frac{(-22) - x}{(-22) - (-11,5)}, & (-11,5) \geq x \geq (-22) \end{cases}$$

$$\mu_{DownHigh}[x] = \begin{cases} 0, & x \geq (-21) \\ \frac{x - (-21)}{(-37,68) - (-21)}, & (-21) \geq x \leq (-37,68) \\ 1, & x \leq (-37,68) \end{cases}$$

$$\mu_{UpLow}[x] = \begin{cases} \frac{3 - x}{3 - 0}, & 0 \leq x \leq 3 \\ 0, & x \geq 3 \end{cases}$$

$$\mu_{UpMid}[x] = \begin{cases} 0, & x \leq 1 \text{ atau } x \geq 22 \\ \frac{x - 1}{11,5 - 1}, & 1 \leq x \leq 11,5 \\ \frac{22 - x}{22 - 11,5}, & 11,5 \leq x \leq 22 \end{cases}$$

$$\mu_{UpHigh}[x] = \begin{cases} 0, & x \leq 21 \\ \frac{x - 21}{32,51 - 21}, & 21 \leq x \leq 32,51 \\ 1, & x \geq 32,51 \end{cases}$$

- Fungsi Keanggotaan JPY/USD

$$\mu_{DownLow}[x] = \begin{cases} \frac{(-1,3) - x}{(-1,3) - 0}, & 0 \geq x \geq (-1,3) \\ 0, & x \leq (-1,3) \end{cases}$$

$$\mu_{DownMid}[x] = \begin{cases} 0, & x \geq -1,1 \text{ atau } x \leq -3,3 \\ \frac{x - (-1,1)}{(-2,2) - (-1,1)}, & -1,1 \geq x \geq (-2,2) \\ \frac{(-3,3) - x}{(-3,3) - (-3,3)}, & (-2,2) \geq x \geq (-3,3) \end{cases}$$

$$\mu_{DownHigh}[x] = \begin{cases} 0, & x \geq (-3,2) \\ \frac{x - (-3,2)}{(-10) - (-3,2)}, & (-3,2) \geq x \geq (-10) \\ 1, & x \leq (-10) \end{cases}$$

$$\mu_{UpLow}[x] = \begin{cases} \frac{1,3 - x}{1,3 - 0}, & 0 \leq x \leq 1,3 \\ 0, & x \geq 1,3 \end{cases}$$

$$\mu_{UpMid}[x] = \begin{cases} 0, & x \leq 1,1 \text{ atau } x \geq 3,3 \\ \frac{x - 1,1}{2,2 - 1,1}, & 1,1 \leq x \leq 2,2 \\ \frac{3,3 - x}{3,3 - 2,2}, & 2,2 \leq x \leq 3,3 \end{cases}$$

$$\mu_{UpHigh}[x] = \begin{cases} 0, & x \leq 3,3 \\ \frac{x - 3,3}{10 - 3,3}, & 3,3 \leq x \leq 10 \\ 1, & x \geq 10 \end{cases}$$

- Fungsi Keanggotaan EUR/USD

$$\mu_{DownLow}[x] = \begin{cases} \frac{(-4) - x}{(-4) - 0}, & 0 \geq x \geq (-4) \\ 0, & x \leq (-4) \end{cases}$$

$$\mu_{DownMid}[x] = \begin{cases} 0, & x \geq -2 \text{ atau } x \leq -22 \\ \frac{x - (-2)}{(-11) - (-2)}, & -2 \geq x \geq (-11) \\ \frac{(-20) - x}{(-20) - (-11)}, & (-11) \geq x \geq (-20) \end{cases}$$

$$\mu_{DownHigh}[x] = \begin{cases} 0, & x \geq (-19) \\ \frac{x - (-19)}{(-100) - (-19)}, & (-19) \geq x \geq (-100) \\ 1, & x \leq (-100) \end{cases}$$

$$\mu_{UpLow}[x] = \begin{cases} \frac{4 - x}{4 - 0}, & 0 \leq x \leq 4 \\ 0, & x \geq 4 \end{cases}$$

$$\mu_{UpMid}[x] = \begin{cases} 0, & x \leq 2 \text{ atau } x \geq 22 \\ \frac{x - 2}{11 - 2}, & 2 \leq x \leq 11 \\ \frac{20 - x}{20 - 11}, & 11 \leq x \leq 20 \end{cases}$$

$$\mu_{UpHigh}[x] = \begin{cases} 0, & x \leq 19 \\ \frac{x - 19}{100 - 19}, & 19 \leq x \leq 100 \\ 1, & x \geq 100 \end{cases}$$

4.4.1.3 Menghitung Nilai Keanggotaan

Kemudian menghitung nilai keanggotaan berdasarkan fungsi keanggotaan yang telah dijelaskan sebelumnya. Perhitungan ini adalah menghitung nilai keanggotaan dalam tiap-tiap fungsi keanggotaan berdasarkan nilai variabel. Perhitungan untuk tiap variabel dapat dicontohkan sebagai berikut.

- Nilai Keanggotaan USD/GBP

$$\mu_{DownLow}[8,95] = 0$$

$$\mu_{DownMid}[8,95] = 0$$

$$\mu_{DownHigh}[8,95] = 0$$

$$\mu_{UpLow}[8,95] = 0$$

$$\mu_{UpMid}[8,95] = \frac{8,95 - 6}{17,5 - 6} = 0,257$$

$$\mu_{UpHigh}[8,95] = 0$$

- Nilai Keanggotaan CHF/USD

$$\mu_{DownLow}[10,79] = 0$$

$$\mu_{DownMid}[10,79] = 0$$

$$\mu_{DownHigh}[10,79] = 0$$

$$\mu_{UpLow}[10,79] = 0$$

$$\mu_{UpMid}[10,79] = \frac{10,79 - 1}{11,5 - 1} = 0,926$$

$$\mu_{UpHigh}[10,79] = 0$$

- Nilai Keanggotaan JPY/USD

$$\mu_{DownLow}[-0,79] = \frac{(-1,3) - (-0,79)}{(-1,3) - 0} = 0,386$$

$$\mu_{DownMid}[-0,79] = 0$$

$$\mu_{DownHigh}[-0,79] = 0$$

$$\mu_{UpLow}[-0,79] = 0$$

$$\mu_{UpMid}[-0,79] = 0$$

$$\mu_{UpHigh}[-0,79] = 0$$

- Nilai Keanggotaan EUR/USD

$$\mu_{DownLow}[13,44] = 0$$

$$\mu_{DownMid}[13,44] = 0$$

$$\mu_{DownHigh}[13,44] = 0$$

$$\mu_{UpLow}[13,44] = 0$$

$$\mu_{UpMid}[13,44] = \frac{20,0 - 13,44}{20,0 - 11} = 0,728$$

$$\mu_{UpHigh}[13,44] = 0$$

4.4.1.4 Inferensi

Setelah nilai keanggotaan didapatkan selanjutnya adalah mencari inferensi. Dalam proses inferensi ini mencari nilai minimal (α -predikat) tiap nilai keanggotaan berdasarkan rule IF-THEN yang ada. Kemudian mencari nilai Z berdasarkan rule tersebut. Dalam kasus ini nilai Z adalah harga emas yang akan diramal. Perhitungan nilai Z menggunakan Persamaan 2.1. Pada contoh proses perhitungan inferensi ini hanya menggunakan 2 rule dari 30 rule yang ada. Perhitungannya adalah sebagai berikut:

[R1] IF USD/GBP UPLOW and CHF/USD UPLOW and JPY/USD UPLOW and EUR/USD UPLOW;

$$\alpha\text{-predikat}_1 = \min(\mu_{USD/GBP \text{ UPLOW}} \cap \mu_{CHF/USD \text{ UPLOW}} \cap \mu_{JPY/USD \text{ UPLOW}} \cap \mu_{EUR/USD \text{ UPLOW}})$$

$$= \min(0 \cap 0,48287 \cap 0,763315 \cap 0)$$

$$= 0$$

THEN Gold UPLOW

$$Z = 12 - 0 \times (12-0)$$

$$Z = 12$$

[R2] IF USD/GBP UPHIGH and CHF/USD UPHIGH and JPY/USD UPHIGH and EUR/USD UPHIGH;

$$\alpha\text{-predikat}_2 = \min(\mu_{USD/GBP \text{ UPHIGH}} \cap \mu_{CHF/USD \text{ UPHIGH}} \cap \mu_{JPY/USD \text{ UPHIGH}} \cap \mu_{EUR/USD \text{ UPHIGH}})$$

$$= \min(0 \cap 0 \cap 0 \cap 0)$$

$$= -0$$

THEN Gold UPHIGH

$$Z = 100 - 0 \times (100-44)$$

$$Z = 100$$

4.4.1.5 Defuzzifikasi

Proses Defuzzifikasi adalah proses dimana nilai α -predikat dikalikan dengan nilai Z lalu di totalkan. Hasil perkalian ini kemudian dibagi dengan jumlah semua α -predikat untuk mendapatkan nilai Z final. Untuk persamaan defuzzifikasi dapat dilihat pada Persamaan 2.2. Sebagai contoh untuk data pada tanggal 10 Januari 2013 didapatkan jumlah nilai α -predikat dikali nilai Z pada proses inferensi sebesar 1,307 dan jumlah semua α -predikat adalah 0,173.

$$Z = \frac{1,307}{0,173} = 7,537$$

Dari perhitungan diatas dapat dilihat bahwa nilai defuzzifikasi yang dihasilkan adalah sebesar 7,357. Ini artinya sistem meramalkan bahwa harga emas pada tanggal 11 Januari 2013 akan naik sebesar 7,537.

4.4.2 Menghitung Nilai Error

Setelah hasil perhitungan defuzzifikasi didapatkan, langkah selanjutnya adalah menghitung nilai *error* dari perhitungan tersebut. Caranya adalah dengan mengurangi nilai aktual dengan nilai Z_{final} . Setelah itu hasilnya akan dikuadratkan. Hasil error kuadrat akan dijumlahkan, lalu diakarkan untuk mendapatkan nilai RMSE. Rumus baku pada perhitungan RMSE ini dapat dilihat pada Persamaan 2.4. Nilai Total, dan RMSE yang tercantum didapatkan dari perhitungan menggunakan semua data yang ada.

$$RMSE = \sqrt{\frac{180532,7}{1000}} = 13,43625$$

4.4.3 Algoritme Genetika

Setelah mendapatkan hasil dari FIS Tsukamoto selanjutnya akan dioptimasi dengan menggunakan Algoritme Genetika. Setiap fungsi keanggotaan akan dijadikan kromosom, kemudian dibangkitkan sebanyak populasi untuk selanjutnya dihitung dengan Algoritme Genetika untuk mendapatkan fungsi keanggotaan dengan *fitness* terbaik.

4.4.3.1 Representasi Kromosom

Pada representasi kromosom fungsi keanggotaan setiap variabel akan direpresentasikan sebagai kromosom. Representasi kromosom yang digunakan adalah representasi *double*, representasi *double* merupakan representasi angka dalam bentuk desimal. Representasi *double* digunakan karena data yang digunakan dalam bentuk desimal sehingga setiap gen dalam kromosom direpresentasikan dalam bentuk nilai desimal pula. Representasi kromosom ini terdiri dari 5 variabel dan setiap variabel memiliki 4 gen. Sehingga total panjang kromosom adalah 20. Setiap angka dalam gen akan diacak dengan rentang antara 0 sampai nilai maksimum pada setiap variabel. Contoh representasi kromosom penelitian ini dapat dilihat pada Tabel 4.3.

Tabel 4.3 Representasi Kromosom

Parent	USD/GBP				CHF/USD				JPY/USD	
	A1	A2	A3	A4	B1	B2	B3	B4	C1	C2
P1	6	8	29	30	1	3	21	22	1,1	1,3
P2	9	11	31	32	6	7	24	25	1,5	1,6
P3	12	14	34	35	9	10	27	28	1,7	1,9

Parent	JPY/USD		EUR/USD				GOLD			
	C3	C4	D1	D2	D3	D4	E1	E2	E3	E4
P1	3,2	3,3	2	4	19	20	11	12	44	45
P2	2,7	3,2	5	7	21	23	14	15	47	48
P3	2,8	3,4	9	11	24	25	17	18	50	51

4.4.3.2 Fitness

Nilai *fitness* adalah faktor yang sangat penting dalam menentukan solusi pada permasalahan optimasi. Perhitungan *fitness* ini didapatkan dengan menghitung peramalan tiap fungsi keanggotaan, kemudian akan dicari nilai RMSE setiap individu yang telah dibahas sebelumnya. Nilai RMSE inilah yang digunakan untuk mendapatkan nilai *fitness* setiap individu dengan cara membagi konstanta dengan nilai RMSE. Dalam penelitian ini, Nilai konstanta yang digunakan adalah 100. Perhitungan nilai *fitness* ini akan dilakukan dengan menggunakan Persamaan 2.4.

4.4.3.3 Crossover

Langkah selanjutnya adalah melakukan reproduksi dengan menggunakan *crossover*. Meode ini dilakukan untuk menghasilkan keturunan dari populasi. Metode *crossover* yang digunakan adalah *one-cut point*. Contoh kerja operator ini adalah dengan memilih gen tertentu secara acak. Gen ini kemudian dijadikan titik poin untuk *crossover* (*cut point*). Kromosom baru pertama berisi gen pertama sampai gen *cut point* dari kromosom induk pertama ditambah dengan gen dari *cut point* sampai gen terakhir dari kromosom induk kedua. Kromosom baru kedua berisi gen pertama sampai gen *cut point* dari induk kedua ditambahkan dengan gen dari *cut point* sampai gen dari kromosom induk pertama.

Tabel 4.4 Crossover One Cut Point

Parent	A1	A2	A3	A4	B1	B2	B3	B4	C1	C2
P1	6	8	29	30	1	3	21	22	1,1	1,3
P2	9	11	31	32	6	7	24	25	1,5	1,6
C1	6	8	29	30	1	3	21	22	1,5	1,6
C2	9	11	31	32	6	7	24	25	1,1	1,3

Parent	C3	C4	D1	D2	D3	D4	E1	E2	E3	E4
P1	3,2	3,3	2	4	19	20	11	12	44	45
P2	2,7	3,2	5	7	21	23	14	15	47	48
C1	2,7	3,2	5	7	21	23	14	15	47	48
C2	3,2	3,3	2	4	19	20	11	12	44	45

4.4.3.4 Mutasi

Proses mutasi penelitian ini menggunakan *random mutation*. Proses ini dilakukan dengan memilih satu induk secara acak dari populasi. Dari induk ini kemudian akan dipilih lagi satu gen untuk dilakukan mutasi. Kemudian gen yang terpilih tersebut akan ditambah atau dikurangi dengan bilangan random yang lebih kecil. Misalkan domain variabel x_j adalah $[min_j, max_j]$ dan offspring yang dihasilkan adalah $C = [x'_j, x'_j]$, maka nilai gen offspring akan dibangkitkan dengan menggunakan Persamaan 2.3 dimana nilai r yang digunakan berada dalam rentang $[0;0,1]$ yaitu, 0,012. Misal gen yang terpilih adalah C2, maka:

$$x'_i = 1,9 + (0,012 (10-0)) = 2,02$$

Tabel 4.5 Mutasi

Parent	A1	A2	A3	A4	B1	B2	B3	B4	C1	C2
P3	12	14	34	35	9	10	27	28	1,7	1,9
C3	12	14	34	35	9	10	27	28	1,7	2,02

Parent	C3	C4	D1	D2	D3	D4	E1	E2	E3	E4
P3	2,8	3,4	9	11	24	25	17	18	50	51
C3	2,8	3,4	9	11	24	25	17	18	50	51

4.4.3.5 Evaluasi

Evaluasi digunakan untuk menghitung kebugaran (*fitness*) setiap kromosom. Semakin besar *fitness* maka semakin baik kromosom tersebut dijadikan calon solusi. Kromosom yang dievaluasi merupakan kromosom gabungan dari populasi awal yang diinisialisasi dan hasil *offspring* yang didapatkan dari proses *crossover* dan mutasi. Setiap kromosom dievaluasi berdasarkan nilai *fitness*-nya, semakin besar nilai *fitness* sebuah kromosom maka akan semakin baik pula solusi yang diberikan. Dari hasil proses evaluasi inilah nantinya akan digunakan pada proses seleksi. Nilai evaluasi dapat dilihat pada Tabel 4.7.

Tabel 4.6 Nilai Evaluasi

Kromosom	Fitnesss
P1	8,2842
P2	8,1576
P3	7,7177
C1	7,6964
C2	7,6949
C3	7,5334

4.4.3.6 Seleksi

Seleksi dilakukan untuk memilih individu dari himpunan populasi dan *offspring* yang dipertahankan hidup pada generasi berikutnya. Semakin besar nilai

fitness sebuah kromosom maka semakin besar peluangnya untuk terpilih. Hal ini bertujuan agar generasi yang terbentuk berikutnya lebih baik dari generasi sekarang.

Penelitian ini menggunakan *elitism selection*. Metode ini memilih individu terbaik dari kumpulan individu di populasi yaitu induk (*parent*) dan *offspring* sejumlah *popSize*. Individu terbaik ini didapatkan dari individu yang menghasilkan *fitness* terbesar. Hasil seleksi individu ini dapat dilihat pada Tabel 4.8.

Tabel 4.7 Seleksi berdasarkan *fitness*

Kromosom baru	Seleksi	Fitnesss
P1	P2	8,2383
P2	C1	7,6784
P3	P3	7,6670
P4	C2	7,6011
P5	C3	7,4124
P6	P1	6,8801

Kromosom P2, P3, dan C3 terpilih menjadi induk baru untuk iterasi kedua. Seleksi ini hanya memilih 3 induk karena *popSize* yang ditentukan adalah 3. Kemudian dilakukan lagi proses *crossover* sampai seleksi ini dan akan menghasilkan induk baru lagi yang memiliki nilai *fitness* tertinggi. Proses *crossover* sampai seleksi ini akan diulang berkali-kali sampai kondisi berhenti.

Tabel 4.8 Seleksi pada Iterasi kedua

Kromosom baru	Seleksi	Fitnesss
P1	P1	8,2842
P2	C1	8,2842
P3	P2	8,1576
P4	C2	8,1576
P5	P3	7,7177
P6	C3	7,7173

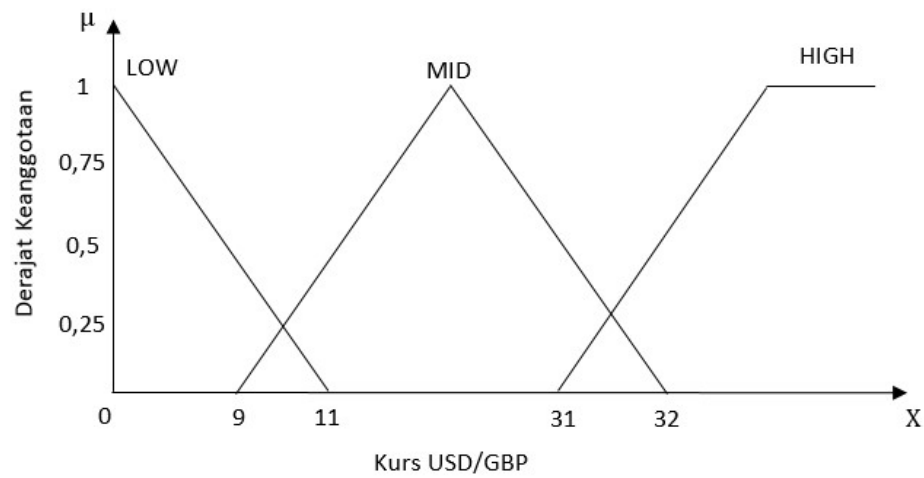
Pada hasil seleksi iterasi kedua maka individu yang terpilih adalah P1 dengan nilai *fitness* 8,2811. Individu ini jika dilakukan proses *decoding* maka setiap gen pada kromosom yang terpilih dapat dilihat pada tabel 4.10. Penjabaran fungsi keanggotannya dapat dilihat pada Gambar 4.17 – Gambar 4.21.

Tabel 4.9 Gen dari individu yang terpilih

A1	A2	A3	A4	B1	B2	B3	B4	C1	C2	C3	C4
9	11	31	32	6	7	24	25	1,5	1,6	2,7	3,2

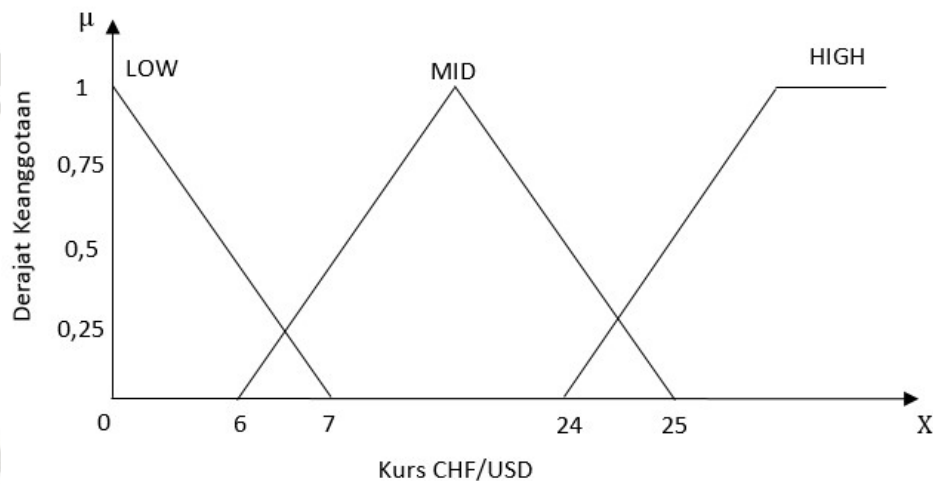
D1	D2	D3	D4	E1	E2	E3	E4
5	7	21	23	14	15	47	48

- USD/GBP



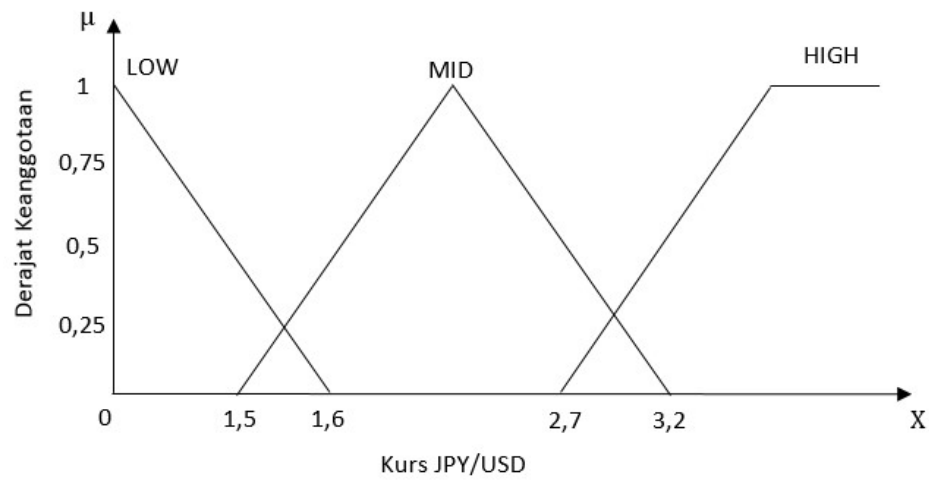
Gambar 4.17 Fungsi Keanggotaan USD/GBP

- CHF/USD



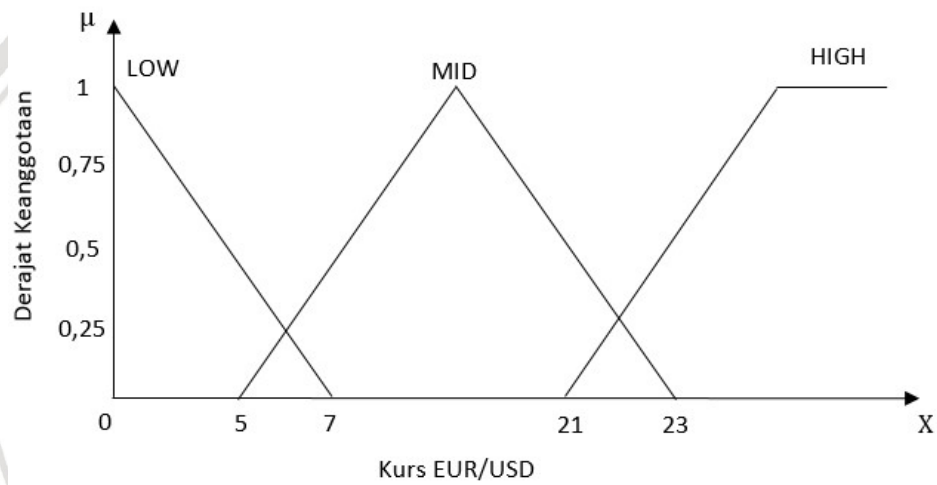
Gambar 4.18 Fungsi Keanggotaan CHF/USD

- JPY/USD



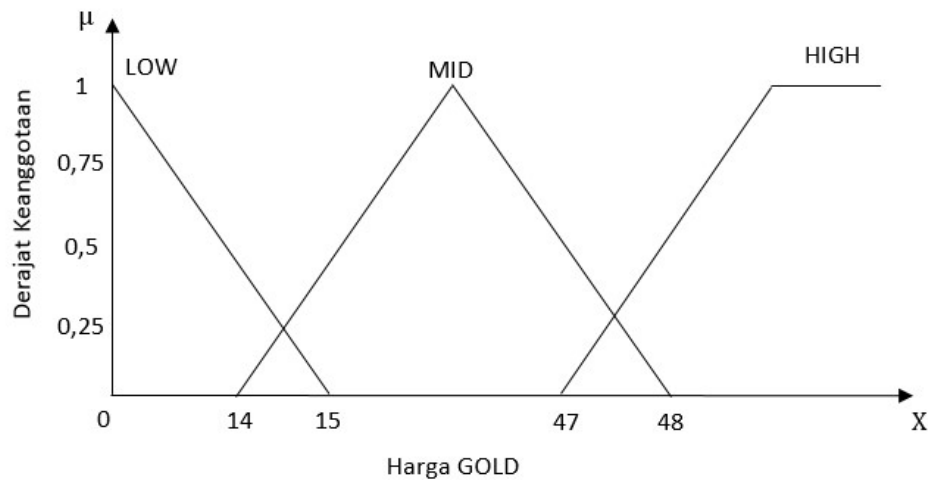
Gambar 4.19 Fungsi Keanggotaan JPY/USD

- EUR/USD



Gambar 4.20 Fungsi Keanggotaan EUR/USD

- GOLD



Gambar 4.21 Fungsi Keanggotaan GOLD

4.5 Analisis Kebutuhan Perangkat Lunak

Analisis kebutuhan perangkat lunak mencakup kegiatan yang diperlukan sistem, antara lain identifikasi pengguna, penjelasan kebutuhan masukan, proses, dan keluaran.

4.5.1 Identifikasi Pengguna

Pada tahap ini direpresentasikan *user* yang menggunakan sistem. Sistem diperuntukkan bagi semua *user* tanpa klasifikasi tingkatan. Semua *user* dapat menggunakan sistem optimasi dan peramalan harga emas ini.

4.5.2 Analisis Kebutuhan Masukan

Masukan yang dibutuhkan sistem untuk optimasi adalah jumlah *popSize*, nilai *cr* dan *mr*, serta jumlah iterasi. Sedangkan untuk peramalan, masukan yang dibutuhkan sistem adalah kurs mata uang USD/GBP, CHF/USD, JPY/USD, dan EUR/USD 3 hari sebelum hari H dan data harga emas 1 hari sebelum hari H. Dari data masukan tersebut, sistem akan memproses perhitungan optimasi dan peramalan dengan kombinasi antara Algoritme Genetika dan FIS Tsukamoto.

4.5.3 Analisis Kebutuhan Keluaran

Ada 2 keluaran yang diharapkan dari sistem ini. Yang pertama adalah fungsi keanggotaan dengan nilai *fitness* terbaik yang didapatkan dengan menggunakan metode Algoritme Genetika. Yang kedua adalah hasil peramalan harga emas. Jika harga emas menurun maka sistem akan menampilkan tanda minus (-) diikuti dengan angka sejumlah penurunan. Dan apabila harga naik maka sistem akan menampilkan angka sejumlah kenaikan. Hasil peramalan sistem kemudian dibandingkan dengan data asli. Perbandingan hasil dari sistem dengan data asli dihitung sehingga didapat tingkat akurasi sistem.

4.6 Perancangan Sistem

Pada tahap ini akan digambarkan pembuatan sistem, yaitu dengan akuisisi pengetahuan dan memodelkan sistem dengan diagram alir atau *flowchart*. Proses yang digambarkan pada perancangan ini yaitu proses sistem secara umum, proses sistem optimasi dengan Algoritme Gentika dan proses peramalan dengan FIS Tsukamoto.

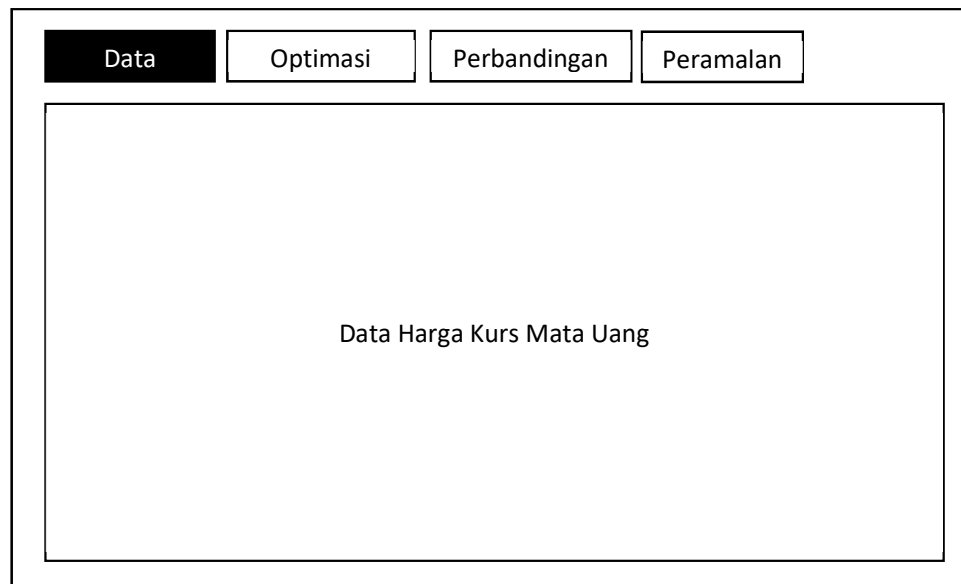
Pada tahap perhitungan nilai keanggotaan data akan dilakukan perhitungan setiap data berdasarkan fungsi keanggotaan yang telah didefinisikan. Kemudian setelah semua data dihitung akan dicari nilai minimum tiap data berdasarkan rule IF-THEN yang ada. Nilai minimum ini disebut dengan α -predikat. Setelah α -predikat didapatkan sistem akan mencari nilai Z dari setiap rule. Kemudian sistem akan mencari jumlah dari α -predikat dikalikan dengan Z dan dibagi dengan total α -predikat. Proses ini disebut dengan defuzzifikasi dan hasil perhitungan pada proses inilah yang menjadi nilai peramalan. Setelah itu akan dilakukan proses optimasi dengan Algoritme Genetika. Hasil peramalan FIS Tsukamoto akan dicari nilai *error*-nya untuk mendapatkan nilai *fitness* setiap fungsi keanggotaan. Setiap fungsi keanggotaan akan direpresentasikan sebagai kromosom dalam Algoritme Genetika. Kromosom ini akan direproduksi dengan proses *crossover* dan mutasi. Selanjutnya dilakukan proses evaluasi dan seleksi guna mendapatkan kromosom dengan nilai *fitness* terbaik.

4.7 Perancangan Antarmuka

Perancangan antarmuka untuk sistem Optimasi Fungsi Keanggotaan FIS Tsukamoto dengan menggunakan Algoritme Genetika pada Peramalan Harga Emas untuk Stock Trading terdapat 4 macam perancangan antarmuka. Perancangan antarmuka ini dibuat berdasarkan fungsi yang akan ditampilkan pada sistem.

4.7.1 Halaman Data

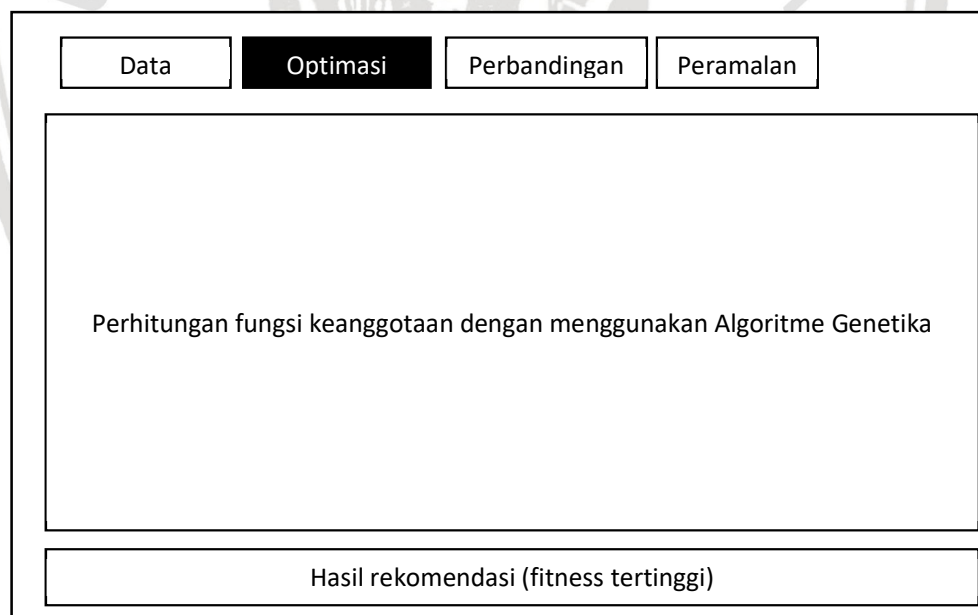
Ketika *user* mengklik navbar Data maka sistem akan langsung menampilkan data dari *dashboard* paling atas yaitu detail data. Di *dashboard* data ini sistem menampilkan data harga kurs mata uang yang digunakan untuk mencari harga emas. Halaman ini dapat dilihat pada Gambar 4.22.



Gambar 4.22 Perancangan Halaman Data

4.7.2 Halaman Optimasi

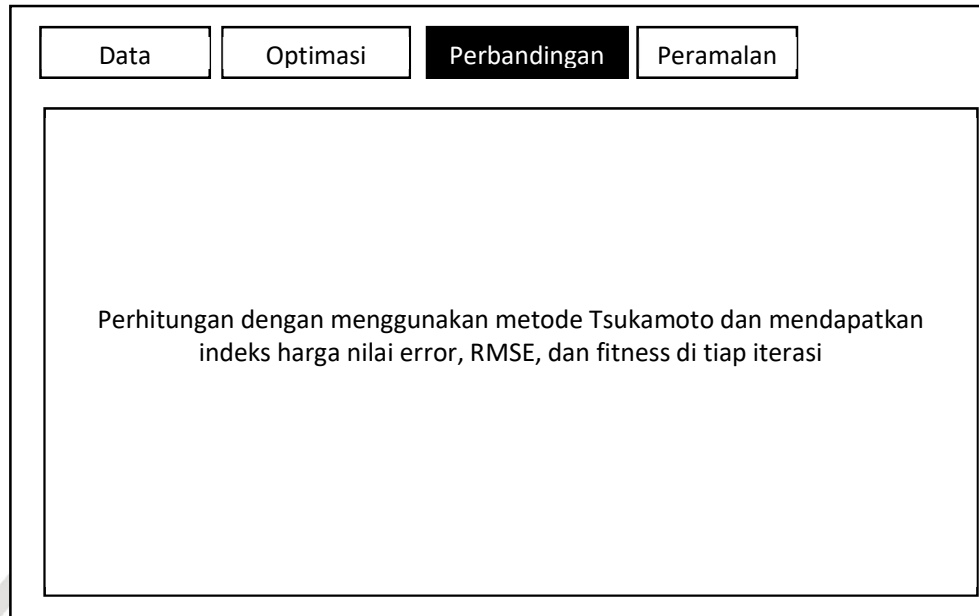
Halaman Optimasi memuat perhitungan optimasi yang dilakukan sistem dengan metode Algoritme Genetika. Pada laman ini *user* akan memasukkan masukan berupa *crossover rate*, *mutation rate*, jumlah populasi dan jumlah iterasi. *User* kemudian akan menekan tombol untuk memulai perhitungan. Ketika sistem sudah selesai menghitung, sistem akan menampilkan individu dengan *fitness* terbaik pada iterasi terakhir. Halaman ini dapat dilihat pada Gambar 4.23.



Gambar 4.23 Perancangan Halaman Optimasi

4.7.3 Halaman Perbandingan

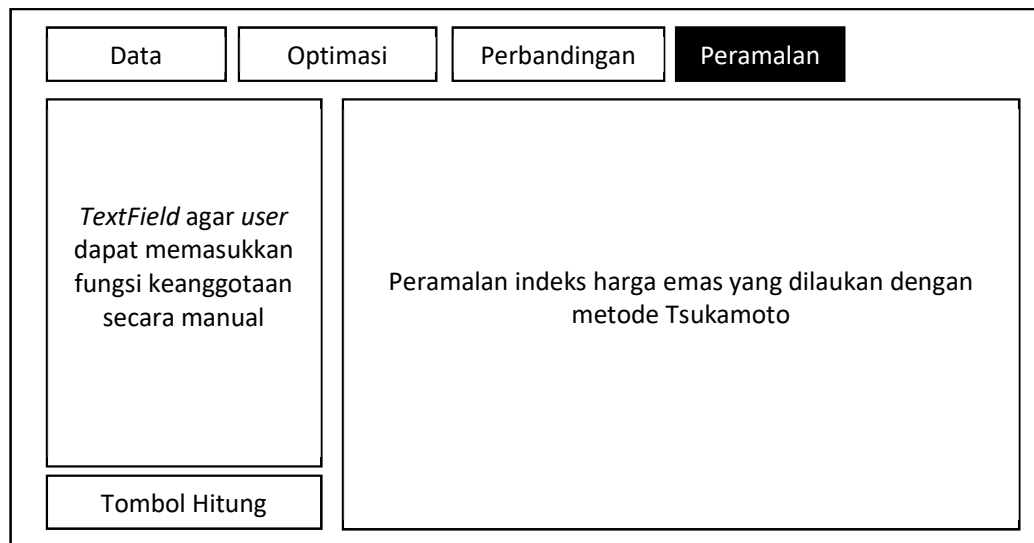
Halaman Perbandingan memuat indeks nilai *error*, RMSE dan *fitness* yang dihasilkan oleh sistem pada tiap iterasi. Halaman ini dapat dilihat pada Gambar 4.24.



Gambar 4.24 Perancangan Halaman Perhitungan

4.7.4 Halaman Peramalan

Halaman Peramalan memuat indeks harga emas sebagai hasil perhitungan yang telah dilakukan dengan metode Tsukamoto. Pada panel di sebelah kanan akan menampilkan indeks harga peramalan menggunakan fungsi keanggotaan berdasarkan hasil rekomendasi yang didapatkan di panel Optimasi. Pada panel ini menampilkan indeks harga peramalan berdasarkan data latih. Panel di sebelah kiri menyediakan tempat untuk *user* memasukkan fungsi keanggotaan secara manual. Setelah tombol Hitung ditekan, maka panel sebelah kanan akan memperbaharui indeks harga peramalan berdasarkan data uji. Halaman ini dapat dilihat pada Gambar 4.25



Gambar 4.25 Perancangan Halaman Peramalan

4.8 Perancangan Pengujian

Pada sub bab ini akan dijelaskan terkait perancangan pengujian data terhadap sistem. Pengujian yang dilakukan antara lain pengujian populasi, pengujian generasi/iterasi, pengujian cr/mr, pengujian dengan parameter terbaik serta pengujian akurasi. Hasil dari proses pengujian ini merupakan hasil kromosom terbaik berdasarkan nilai fitness dan juga akurasi sistem ketika meramalkan harga emas dengan metode tsukamoto. Untuk pengujian akurasi, nilai keluaran dari sistem dikurangi dengan nilai aktual dan dibagi dengan jumlah data.

4.8.1 Pengujian Populasi

Dalam pengujian populasi akan dilakukan masing-masing 10 kali untuk tiap ukuran populasi yang berbeda dengan kenaikan sejumlah 10. Dalam pengujian populasi juga akan digunakan parameter generasi/iterasi, Cr dan Mr . Rancangan pengujian dapat dilihat pada Tabel 4.11.

Tabel 4.10 Perancangan Pengujian Populasi

[illegible]

4.8.2 Pengujian Nilai *Crossover Rate* dan *Mutation Rate*

Dalam pengujian generasi/iterasi akan dilakukan masing-masing 10 kali untuk tiap ukuran generasi yang berbeda dengan kenaikan sejumlah 10. Dalam pengujian generasi juga akan digunakan parameter populasi, *Cr* dan *Mr*. Rancangan pengujian dapat dilihat pada Tabel 4.12.

Tabel 4.11 Perancangan Pengujian Kombinasi Nilai *cr* dan *mr*

<i>cr</i>	<i>mr</i>	Percobaan ke										Rata-rata
		1	2	3	4	5	6	7	8	9	10	
0.1	0.9											
0.2	0.8											
0.3	0.7											
0.4	0.6											
0.5	0.5											
0.6	0.4											
0.7	0.3											
0.8	0.2											
0.9	0.1											

4.8.3 Pengujian Konvergensi

Dalam pengujian konvergensi akan dilakukan masing-masing 10 kali untuk tiap ukuran generasi yang berbeda dengan kenaikan sejumlah 25. Dalam pengujian generasi juga akan digunakan parameter populasi, *Cr* dan *Mr*. Rancangan pengujian dapat dilihat pada Tabel 4.13.

Tabel 4.12 Perancangan Pengujian Konvergensi

Konvergensi	Percobaan ke-										Rata-rata
	1	2	3	4	5	6	7	8	9	10	
25											
50											
75											
100											
125											
150											
175											
200											
...											
400											

4.8.4 Pengujian RMSE (Root Mean Squared Error)

Pengujian RMSE dilakukan setelah optimasi dengan Algoritme Genetika didapatkan. Ketika pengujian dengan Algoritme Genetika selesai maka akan menghasilkan fungsi keanggotaan *fuzzy* dengan nilai *fitness* terbaik. Fungsi keanggotaan ini kemudian akan digunakan untuk menghitung peramalan dengan

metode Tsukamoto dengan menggunakan 100 data uji yang mana hasilnya kemudian akan dihitung tingkat akurasinya. Pengujian RMSE dilakukan dengan menggunakan Persamaan 2.5.



BAB 5 IMPLEMENTASI

5.1 Spesifikasi Perangkat

Pada spesifikasi perangkat ini akan menjelaskan tentang spesifikasi dari perangkat yang digunakan untuk melakukan implementasi penelitian ini. Spesifikasi perangkat dibagi menjadi dua yakni spesifikasi *hardware* (perangkat keras) dan *software* (perangkat lunak).

5.1.1 Spesifikasi Perangkat Keras

Pada pengembangan sistem Optimasi Fungsi Keanggotaan FIS Tsukamoto pada Peramalan Harga Emas untuk *Stock Trading*, memanfaatkan komputer dengan spesifikasi *hardware* (perangkat keras) sebagai berikut:

- Processor Intel Core i7 3.80GHz
- Kapasitas Memori (RAM) 8.00 GB
- VGA Card GTX 1050

5.1.2 Spesifikasi Perangkat Lunak

Pada pengembangan sistem Pada pengembangan sistem Optimasi Fungsi Keanggotaan FIS Tsukamoto pada Peramalan Harga Emas untuk *Stock Trading*, memanfaatkan komputer dengan spesifikasi *software* (perangkat lunak) sebagai berikut:

- OS Windows 10 Profesional 64 bit
- Netbeans 8.0.2

5.2 Implementasi Algoritme

Implementasi Optimasi Fungsi Keanggotaan FIS Tsukamoto dengan Menggunakan Algoritme Genetika pada Peramalan Harga Emas untuk *Stock Trading* ini menggunakan bahasa Java. Implementasi algoritme ini mengacu pada bab perancangan proses dan meliputi proses manualisasi yang telah dijelaskan sebelumnya.

5.2.1 Implementasi Proses Inisialisasi Kromosom

Inisialisasi kromosom awal merupakan langkah awal dalam Algoritme Genetika. Inisialisasi kromosom ini dilakukan dengan membangkitkan gen tiap variabel sesuai dengan batas yang telah ditentukan. Inisialisasi kromosom ini telah dibahas lebih detil pada Subbab 4.4, proses tersebut telah diimplementasikan pada Kode Program 5.1.

Method inisialisasiKromosom	
1	public void inisialisasiKromosom() {
2	int counter = 0;
3	for (int i = 0; i < this.p.length; i++) {
4	for (int j = 0; j < 5; j++) {
5	for (int k = 0; k < 4; k++) {
6	this.p[i][counter++] = (double) (0 + (int)
7	(Math.random() * ((this.MAX_RANDOM[j] *

```

8         10) - (0) + 1))) / 10;
9     }
10    }
11    counter = 0;
12    }
13 }

```

Kode Program 5.1 Insialisasi Kromosom

Penjelasan Kode Program 5.1:

1. Baris 1 adalah deklarasi method proses inialisasi kromosom.
2. Baris 2-9 adalah proses pembangkitan nilai setiap gen secara *random* sesuai dengan nilai maksimum yang ditentukan.

5.2.2 Implementasi Proses *Crossover*

Crossover merupakan proses pertama pada reproduksi dalam Algoritme Genetika. Proses *crossover* pada penelitian ini menggunakan *One Cut Point*, proses *crossover* ini telah dibahas pada Subbab 4.4.11 secara rinci, dan proses tersebut diimplementasikan pada Kode Program 5.2.

```

Method Crossover
1 public void crossover() {
2     int batasRandomPopulasi = 1;
3     int induk1 = 0 + (int) (Math.random() * (((this.popSize -
4         1) - 0) + 1));
5     int induk2 = 0 + (int) (Math.random() * (((this.popSize -
6         1) - 0) + 1));
7     Random r = new Random();
8     int titikPotong = 1 + (int) (Math.random() * (((20 - 2) -
9         1) + 1));
10    double tempPC[][]= new double[this.pc.length]
11        [this.pc[0].length];
12    for (int i = 0; i < this.pc.length; i++) {
13        if (i % 2 == 0) {
14            System.arraycopy(this.p[induk1], 0, tempPC[0], 0,
15                this.pc[0].length);
16            if(this.pc.length != 1){
17                System.arraycopy(this.p[induk2], 0, tempPC[1], 0,
18                    this.pc[0].length);
19            }
20            for (int j = 0; j < this.pc[0].length; j++) {
21                if (j >= titikPotong) {
22                    double temp1 = this.p[induk2][j];
23                    double temp2 = this.p[induk1][j];
24                    tempPC[0][j] = temp1;
25                    if(this.pc.length != 1){
26                        tempPC[1][j] = temp2;
27                    }
28                }
29            }
30            this.pc[i] = Arrays.copyOfRange(tempPC[0], 0,
31                tempPC[0].length);
32        } else {
33            this.pc[i] = Arrays.copyOfRange(tempPC[1], 0,
34                tempPC[1].length);
35        }
36        if (batasRandomPopulasi == 2) {
37            batasRandomPopulasi = 1;
38            induk1 = 0 + (int) (Math.random() *
39                (((this.popSize - 1) - 0) + 1));
40            induk2 = getRandomWithExclusion(r,0,(this.popSize
41                - 1),induk1);
42            titikPotong = 1 + (int) (Math.random() * (((20 -

```

```

43         2) - 1) + 1));
44     } else { batasRandomPopulasi++;
45     }
46 }
47 }

```

Kode Program 5.2 Proses Crossover

Penjelasan Kode Program:

1. Baris 1 adalah deklarasi method proses *crossover*.
2. Baris 2-9 adalah pemilihan individu dan titik potong secara *random* untuk digunakan pada proses *crossover*.
3. Baris 10-35 adalah proses reproduksi menggunakan *One Cut Point*. Sistem akan menentukan dimana titik potongnya. Kemudian, hasil reproduksi setiap anak hasil *crossover* akan disimpan.
4. Baris 36-47 adalah proses pengecekan apabila proses *crossover* menghasilkan lebih dari 1 *offspring*, maka akan dipilih individu baru lagi sebagai induk secara *random* untuk dilakukan lagi proses *crossover* pada *offspring* selanjutnya dengan langkah yang sama seperti sebelumnya.

5.2.3 Implementasi Proses Mutasi

Proses mutasi merupakan proses kedua pada reproduksi dalam Algoritme Genetika. Proses mutasi pada penelitian ini menggunakan *Random Mutation*. Proses *Random Mutation* ini dilakukan dengan cara memilih satu gen pada kromosom. Gen ini kemudian akan dihitung sesuai dengan persamaan yang telah dijelaskan sebelumnya. Proses mutasi ini telah dibahas pada Subbab 4.4.12 dengan lebih rinci, proses tersebut telah diimplementasikan pada Kode Program 5.3.

```

Method mutasi
1  public void mutasi(){
2      for (int i = 0; i < this.pm.length; i++) {
3          int induk = 0 + (int) (Math.random() * (((this.popSize
4              - 1) - 0) + 1));
5          int pos = 0 + (int) (Math.random() * (((20 - 1) - 0) +
6              1));
7          this.pm[i] = Arrays.copyOfRange(this.p[induk], 0,
8              this.p[i].length);
9          double r = (double) (0 + (int) (Math.random() * ((100 - (-
10              100)+1))))/1000;
11         int tipe = 0;
12         if(pos >= 0 && pos <= 3){
13             tipe = 0;
14         }else if(pos >= 4 && pos <= 7){
15             tipe = 1;
16         }else if(pos >= 8 && pos <= 11){
17             tipe = 2;
18         }else if(pos >= 12 && pos <= 15){
19             tipe = 3;
20         }else if(pos >= 16 && pos <= 19){
21             tipe = 4;
22         }
23         double nilai = this.pm[i][pos];
24         double mut = nilai + r * (this.MAX_RANDOM[tipe] - 0);
25         if(tipe == 0 || tipe == 1 || tipe == 3 || tipe == 4){
26             mut = Math.round(mut);
27         }

```

```

28         if(mut > this.MAX_RANDOM[tipe]){
29             mut = this.MAX_RANDOM[tipe];
30         }
31         this.pm[i][pos] = mut;
32     }
33 }

```

Kode Program 5.3 Proses Mutasi

Penjelasan Kode Program:

1. Baris 1 adalah deklarasi method proses mutasi.
2. Baris 2-10 adalah pemilihan individu, gen yang akan dimutasi, dan nilai *r* secara *random* untuk digunakan pada proses mutasi.
3. Baris 11-22 adalah proses mengklasifikasikan kromosom berdasarkan variabelnya. Proses ini dilakukan karena setiap variabel memiliki nilai maksimum *random* yang berbeda.
4. Baris 23-33 adalah proses perhitungan mutasi dimana nilai *random* dalam proses ini didapatkan berdasarkan nilai maksimum *random* setiap variabel yang telah ditentukan sebelumnya.

5.2.4 Implementasi Proses Evaluasi

Proses evaluasi adalah proses menggabungkan semua kromosom dari populasi awal, dan populasi hasil reproduksi. Setelah semua digabungkan kemudian akan ditampilkan semua individu beserta nilai *fitness*-nya. Proses evaluasi ini telah dibahas secara lengkap pada Subbab 4.4.13, proses tersebut telah diimplementasikan pada Kode Program 5.4.

```

Method evaluasi
1 public void evaluasi(){
2     System.arraycopy(this.p, 0, this.ps, 0, this.popSize);
3     System.arraycopy(this.pc, 0, this.ps, this.popSize,
4         this.pc.length);
5     System.arraycopy(this.pm, 0, this.ps, this.popSize +
6         this.pc.length, this.pm.length);
7     for (int i = 0; i < this.ps.length; i++) {
8         double temp[] = sort(this.ps[i]);
9         this.ps[i] = temp;
10    }
11    for (int i = 0; i < this.ps.length; i++) {
12        Tsukamoto t = new Tsukamoto(this.ps[i]);
13        t.prosesTsukamoto ();
14        double hasil[] = t.getHasil();
15        this.ps[i][20] = hasil[0];
16        this.ps[i][21] = hasil[1];
17        this.ps[i][22] = hasil[2];
18    }
19 }
20 public double[] sort(double []arr){
21     Arrays.sort(arr, 0, 4);
22     Arrays.sort(arr, 4, 8);
23     Arrays.sort(arr, 8, 12);
24     Arrays.sort(arr, 12, 16);
25     Arrays.sort(arr, 16, 20);
26     return arr;
27 }

```

Kode Program 5.4 Proses Evaluasi

Penjelasan Kode Program:

1. Baris 1 adalah deklarasi method proses evaluasi.
2. Baris 2-5 adalah menyalin larik individu populasi awal, *offspring* hasil *crosscover*, dan hasil mutasi.
3. Baris 7-10 adalah proses menyortir setiap indeks individu berdasarkan nominal paling kecil sampai paling besar.
4. Baris 12-20 adalah proses menghitung nilai *fitness* dimana proses lengkapnya terdapat pada kelas Tsukamoto.java. Kemudian hasil perhitungan *fitness* ini akan disimpan untuk kemudian ditampilkan.
5. Baris 21-28 adalah penjabaran proses sortir yang mana proses ini menyortir angka pada setiap 4 gen.

5.2.5 Implementasi Proses Seleksi

Proses seleksi merupakan proses untuk menyeleksi setiap individu. Proses seleksi ini akan mengurutkan individu berdasarkan nilai *fitness* yang terbaik. Proses ini telah dibahas dengan lengkap pada Subbab 4.4.14, proses tersebut telah diimplementasikan pada Kode Program 5.5.

Method seleksi	
1	public void seleksi(){
2	double tempSeleksi[][] = new double[this.ps.length]
3	[this.ps[0].length];
4	double hasilSeleksi[][] = new double[this.popSize]
5	[this.ps[0].length];
6	System.arraycopy(this.ps,0,tempSeleksi,0,this.ps.length);
7	tempSeleksi = elitism(tempSeleksi);
8	System.arraycopy(tempSeleksi,0,hasilSeleksi,0,this.popSize);
9	updatePopulasi(hasilSeleksi);
10	}
11	public double[][] elitism(double[][] pSeleksi) {
12	double[][] tempP = pSeleksi;
13	Arrays.sort(tempP, new java.util.Comparator<double[]>() {
14	public int compare(double[] a, double[] b) {
15	return Double.compare(b[b.length - 1], a[a.length
16	- 1]);
17	}
18	});
19	return tempP;
20	}

Kode Program 5.5 Proses Seleksi

Penjelasan Kode Program:

1. Baris 1 adalah deklarasi method proses seleksi.
2. Baris 2-20 merupakan proses seleksi untuk menyeleksi individu yang terbaik sebanyak populasi awal berdasarkan nilai *fitness* sebagai solusi dari iterasi ke-n.

5.2.6 Implementasi Perhitungan Nilai Keanggotaan

Menghitung nilai keanggotaan adalah proses menghitung nilai keanggotaan setiap data berdasarkan rumus fungsi keanggotaan yang sesuai dengan data tersebut. Terdapat enam fungsi keanggotaan pada setiap variabel yaitu DownLow,

DownMid, DownHigh, UpLow, UpMid, dan UpHigh. Proses menghitung nilai keanggotaan ini telah dibahas pada Subbab 4.4.2 secara rinci, dan proses tersebut diimplementasikan pada Kode Program 5.6.

```

Method FuzzyLogic
1  public void prosesFuzzyLogic() {
2      int counter = 0;
3      int indeksIndividu = 0;
4      for (int i = 0; i < this.fl.length; i++) {
5          for (int j = 0; j < 4; j++) {
6              double med = (this.individu[indeksIndividu]+
7                  this.individu[indeksIndividu + 3]) / 2;
8              this.fl[i][counter]=
9                  downLow(this.individu[indeksIndividu+1],
10                      this.DATASET[i][j]);
11              this.fl[i][counter+1]=
12                  downMed(this.individu[indeksIndividu],
13                      this.individu[indeksIndividu + 3],
14                      this.DATASET[i][j], med);
15              this.fl[i][counter+2]=
16                  downHigh(this.individu[indeksIndividu+2],
17                      this.DATASET[i][j],this.BATAS[j]);
18              this.fl[i][counter+3]=
19                  upLow(this.individu[indeksIndividu+1],
20                      this.DATASET[i][j]);
21              this.fl[i][counter+4]=
22                  upMed(this.individu[indeksIndividu],
23                      this.individu[indeksIndividu + 3],
24                      this.DATASET[i][j], med);
25              this.fl[i][counter+5]=
26                  upHigh(this.individu[indeksIndividu + 2],
27                      this.DATASET[i][j],this.BATAS[j]);
28              counter += 6;
29              indeksIndividu += 4;
30          }
31          counter = 0;
32          indeksIndividu = 0;
33      }
34  }
35  public double downLow(double a, double x) {
36      double hasil = 0;
37      if (x <= 0 && x >= (-a)) {
38          hasil = ((-a) - x) / ((-a) - 0);
39      } else if (x <= (-a)) {
40          hasil = 0;
41      }
42      return hasil;
43  }
44  public double downMed(double a, double b, double x, double med) {
45      double hasil = 0;
46      if (x >= -a || x <= -b) {
47          hasil = 0;
48      } else if (x >= -med && x <= -a) {
49          hasil = (x - (-a)) / ((-med) - (-a));
50      } else if (x >= -b && x <= -med) {
51          hasil = ((-b) - x) / ((-b) - (-med));
52      }
53      return hasil;
54  }
55  public double downHigh(double a, double x, double batas) {
56      double hasil = 0;
57      if (x >= (-a)) {
58          hasil = 0;
59      } else if ((-a) >= x && (-batas) <= x) {
60          hasil = (x - (-a)) / ((-batas) - (-a));

```

```

61     } else if (x <= (-batas)) {
62         hasil = 1;
63     }
64     return hasil;
65 }
66 public double upLow(double a, double x) {
67     double hasil = 0;
68     if (x >= 0 && x <= a) {
69         hasil = (a - x) / (a - 0);
70     } else if (x >= a) {
71         hasil = 0;
72     }
73     return hasil;
74 }
75 public double upMed(double a, double b, double x, double med) {
76     double hasil = 0;
77     if (x <= a || x >= b) {
78         hasil = 0;
79     } else if (x >= a && x <= med) {
80         hasil = (x - a) / (med - a);
81     } else if (x >= med && x <= b) {
82         hasil = (b - x) / (b - med);
83     }
84     return hasil;
85 }
86 public double upHigh(double a, double x, double batas) {
87     double hasil = 0;
88     if (x <= a) {
89         hasil = 0;
90     } else if (x >= a && x <= batas) {
91         hasil = (x - a) / (batas - a);
92     } else if (x >= batas) {
93         hasil = 1;
94     }
95     return hasil;
96 }

```

Kode Program 5.6 Proses Perhitungan Nilai Keanggotaan

Penjelasan Kode Program:

1. Baris 1 adalah deklarasi method proses menghitung nilai keanggotaan.
2. Baris 2-34 adalah proses untuk mengambil indeks angka fungsi keanggotaan yang ada pada kromosom Algoritme Genetika untuk kemudian dihitung nilai keanggotaannya.
3. Baris 35-96 adalah proses perhitungan setiap nilai keanggotaan setiap data pada semua variabel. Ada enam fungsi keanggotaan yang digunakan yaitu Down Low, Down Med, Down High, Up Low, Up Med, dan Up High.

5.2.7 Implementasi Perhitungan α -Predikat

Proses perhitungan α -predikat adalah sebuah proses untuk mencari nilai minimal dari nilai keanggotaan variabel berdasarkan *rule* IF yang telah ditentukan sebelumnya. Nilai α -predikat ini akan digunakan dalam proses inferensi. Setelah itu, akan digunakan dalam proses defuzzifikasi. Proses perhitungan α -predikat ini telah dibahas dengan lengkap pada Subbab 4.4.5, proses tersebut telah diimplementasikan pada Kode Program 5.7.

Method Alpha	
1	public void prosesAlpha(){
2	for (int i = 0; i < this.a.length; i++) {
3	this.a[i][0] = findMin(this.fl[i][3], this.fl[i][9],

```

4         this.fl[i][15], this.fl[i][21]);
5     this.a[i][1] = findMin(this.fl[i][5], this.fl[i][11],
6         this.fl[i][17], this.fl[i][23]);
7     this.a[i][2] = findMin(this.fl[i][4], this.fl[i][10],
8         this.fl[i][16], this.fl[i][22]);
9     this.a[i][3] = findMin(this.fl[i][0], this.fl[i][6],
10        this.fl[i][12], this.fl[i][18]);
11    this.a[i][4] = findMin(this.fl[i][1], this.fl[i][7],
12        this.fl[i][13], this.fl[i][19]);
13    this.a[i][5] = findMin(this.fl[i][2], this.fl[i][8],
14        this.fl[i][14], this.fl[i][20]);
15    this.a[i][6] = findMin(this.fl[i][3], this.fl[i][6],
16        this.fl[i][12], this.fl[i][18]);
17    this.a[i][7] = findMin(this.fl[i][5], this.fl[i][7],
18        this.fl[i][13], this.fl[i][19]);
19    this.a[i][8] = findMin(this.fl[i][4], this.fl[i][8],
20        this.fl[i][14], this.fl[i][20]);
21    this.a[i][9] = findMin(this.fl[i][0], this.fl[i][9],
22        this.fl[i][15], this.fl[i][21]);
23    this.a[i][10] = findMin(this.fl[i][1], this.fl[i][11],
24        this.fl[i][17], this.fl[i][23]);
25    this.a[i][11] = findMin(this.fl[i][2], this.fl[i][10],
26        this.fl[i][16], this.fl[i][22]);
27    this.a[i][12] = findMin(this.fl[i][3], this.fl[i][9],
28        this.fl[i][15], this.fl[i][21]);
29    this.a[i][13] = findMin(this.fl[i][5], this.fl[i][11],
30        this.fl[i][17], this.fl[i][23]);
31    this.a[i][14] = findMin(this.fl[i][4], this.fl[i][10],
32        this.fl[i][16], this.fl[i][22]);
33    this.a[i][15] = findMin(this.fl[i][0], this.fl[i][6],
34        this.fl[i][12], this.fl[i][18]);
35    this.a[i][16] = findMin(this.fl[i][1], this.fl[i][7],
36        this.fl[i][13], this.fl[i][19]);
37    this.a[i][17] = findMin(this.fl[i][2], this.fl[i][8],
38        this.fl[i][14], this.fl[i][20]);
39    this.a[i][18] = findMin(this.fl[i][0], this.fl[i][6],
40        this.fl[i][12], this.fl[i][18]);
41    this.a[i][19] = findMin(this.fl[i][1], this.fl[i][7],
42        this.fl[i][13], this.fl[i][19]);
43    this.a[i][20] = findMin(this.fl[i][2], this.fl[i][8],
44        this.fl[i][14], this.fl[i][20]);
45    this.a[i][21] = findMin(this.fl[i][3], this.fl[i][9],
46        this.fl[i][15], this.fl[i][21]);
47    this.a[i][22] = findMin(this.fl[i][5], this.fl[i][11],
48        this.fl[i][17], this.fl[i][23]);
49    this.a[i][23] = findMin(this.fl[i][4], this.fl[i][10],
50        this.fl[i][16], this.fl[i][22]);
51    this.a[i][24] = findMin(this.fl[i][4], this.fl[i][10],
52        this.fl[i][16], this.fl[i][22]);
53    this.a[i][25] = findMin(this.fl[i][4], this.fl[i][10],
54        this.fl[i][16], this.fl[i][22]);
55    this.a[i][26] = findMin(this.fl[i][4], this.fl[i][10],
56        this.fl[i][16], this.fl[i][22]);
57    this.a[i][27] = findMin(this.fl[i][1], this.fl[i][7],
58        this.fl[i][13], this.fl[i][19]);
59    this.a[i][28] = findMin(this.fl[i][1], this.fl[i][7],
60        this.fl[i][13], this.fl[i][19]);
61    this.a[i][29] = findMin(this.fl[i][1], this.fl[i][7],
62        this.fl[i][13], this.fl[i][19]);
63    }
64    public double findMin(double a, double b, double c, double d) {
65        double min[] = {a, b, c, d};
66        double minValue = min[0];
67        for (int i = 0; i < min.length; i++) {
68            if (min[i] < minValue) {

```

69	}
70	}
71	return minValue;
72	}

Kode Program 5.7 Proses Perhitungan α -Predikat

Penjelasan Kode Program:

1. Baris 1 adalah deklarasi method proses perhitungan alfa-predikat.
2. Baris 2-63 adalah proses mencari nilai minimal dari tiap nilai keanggotaan berdasarkan *rule* IF yang ada.
3. Baris 64-72 adalah proses perhitungan untuk mencari nilai minimal yang akan digunakan.

5.2.8 Implementasi Perhitungan Nilai Z

Proses perhitungan nilai Z adalah proses untuk mencari nilai Z berdasarkan *rule* THEN yang ada. Proses ini akan menggunakan nilai Z maksimum dan nilai Z minimum, yang mana nilai ini didapatkan dari fungsi keanggotaan variabel GOLD. Nilai Z ini juga termasuk dalam proses inferensi akan digunakan dalam proses defuzzifikasi. Proses ini telah dibahas dengan lengkap pada Subbab 4.4.12, proses tersebut telah diimplementasikan pada Kode Program 5.8.

Method Alpha	
1	public void prosesZ() {
2	for (int i = 0; i < this.z.length; i++) {
3	this.z[i][0]=goldUp(this.a[i][0],0,this.individu[17]);
4	this.z[i][1]=goldUp(this.a[i][1],this.individu[18],100);
5	this.z[i][2]=goldUp(this.a[i][2],this.individu[16],
6	this.individu[19]);
7	this.z[i][3]=goldDown(this.a[i][3],0,this.individu[17]);
8	this.z[i][4]=goldDown(this.a[i][4],this.individu[16],
9	this.individu[19]);
10	this.z[i][5]=goldDown(this.a[i][5],this.individu[18],100);
11	this.z[i][6]=0;
12	this.z[i][7]=goldUp(this.a[i][7],this.individu[16],
13	this.individu[19]);
14	this.z[i][8]=goldUp(this.a[i][8], 0, this.individu[17]);
15	this.z[i][9]=0;
16	this.z[i][10]=goldDown(this.a[i][10],0,this.individu[17]);
17	this.z[i][11]=goldDown(this.a[i][11],this.individu[18],
18	100);
19	this.z[i][12]=goldUp(this.a[i][12],0,this.individu[17]);
20	this.z[i][13]=goldUp(this.a[i][13],this.individu[18],100);
20	this.z[i][14]=goldUp(this.a[i][14],0,this.individu[17]);
21	this.z[i][15]=goldDown(this.a[i][15],0,this.individu[17]);
22	this.z[i][16]=goldDown(this.a[i][16],this.individu[18],
23	100);
24	this.z[i][17]=goldDown(this.a[i][17],this.individu[18],
25	100);
26	this.z[i][18]=0;
27	this.z[i][19]=goldUp(this.a[i][19],0,this.individu[17]);
28	this.z[i][20]=goldDown(this.a[i][20],this.individu[18],
29	100);
30	this.z[i][21]=0;
31	this.z[i][22]=goldUp(this.a[i][22],0,this.individu[17]);
32	this.z[i][23]=goldUp(this.a[i][23],0,this.individu[17]);
33	this.z[i][24]=goldUp(this.a[i][24],this.individu[16],
34	this.individu[19]);
35	this.z[i][25]=goldUp(this.a[i][25],this.individu[16],
36	this.individu[19]);


```

37      this.z[i][26]=goldUp(this.a[i][26],this.individu[16],
38                          this.individu[19]);
39      this.z[i][27]=goldDown(this.a[i][27],this.individu[16],
40                             this.individu[19]);
41      this.z[i][28]=goldDown(this.a[i][28],this.individu[16],
42                             this.individu[19]);
43      this.z[i][29]=goldDown(this.a[i][29],this.individu[16],
44                             this.individu[19]);
45      }
46  }
47  public double goldDown(double alpha, double Zmin, double Zmax){
48      return (-Zmax) + (alpha * ((-Zmax) + Zmin));
49  }
50  public double goldUp(double alpha, double Zmin, double Zmax){
51      return Zmax - (alpha * (Zmax - Zmin));
52  }

```

Kode Program 5.8 Proses Perhitungan Nilai Z

Penjelasan Kode Program:

1. Baris 1 adalah deklarasi method proses perhitungan nilai Z.
2. Baris 2-46 adalah proses mencari nilai harga emas (nilai Z) berdasarkan *rule* THEN yang ada. Proses ini menggunakan individu dari variabel GOLD.
3. Baris 47-53 adalah proses perhitungan untuk mencari nilai Z yang digunakan. Ada dua kategori yang digunakan yaitu harga emas ketika turun (*Down*) dan harga emas ketika naik (*Up*).

5.2.9 Implementasi Proses Defuzzifikasi dan Perhitungan Fitness

Proses defuzzifikasi adalah tahap akhir dalam metode Tsukamoto. Dalam proses ini, akan menggunakan nilai alfa-predikat dan nilai Z yang telah dihitung sebelumnya. Kedua nilai ini kemudian akan dihitung dengan menggunakan Persamaan 4.2. Setelah hasil defuzzifikasi didapatkan, kemudian akan dilakukan perhitungan *fitness*. Proses defuzzifikasi ini telah dibahas dengan lengkap pada Subbab 4.4.6, sedangkan perhitungan *fitness* telah dibahas pada Subbab 4.4.10 secara rinci. Kedua proses tersebut telah diimplementasikan pada Kode Program 5.9.

Method defuzzifikasi	
1	public void prosesAZ(){
2	double total = 0;
3	for (int i = 0; i < this.a.length; i++) {
4	double sumAZ = 0;
5	double sumA = 0;
6	double zfinal, error, errorKuadrat;
7	for (int j = 0; j < this.a[i].length; j++) {
8	this.az[i][j] = this.a[i][j] * this.z[i][j];
9	sumAZ += this.az[i][j];
10	sumA += this.a[i][j];
11	}
12	if(sumAZ != 0 && sumA != 0){
13	zfinal = sumAZ / sumA;
14	}else{
15	zfinal = 0;
16	}
17	error = this.DATASET[i][4] - zfinal;
18	errorKuadrat = Math.pow(error, 2);
19	total += errorKuadrat;
20	this.az[i][30] = sumAZ;
20	this.az[i][31] = sumA;

```

21         this.az[i][32] = zfinal;
22         this.az[i][33] = error;
23         this.az[i][34] = errorKuadrat;
24     }
25     this.total = total;
26     this.totalerror = totalerror;
27     this.rmse = Math.sqrt(this.total / 1000);
28     this.fitness = 100 / this.rmse;
29 }

```

Kode Program 5.9 Proses Defuzzifikasi dan Perhitungan *Fitness*

Penjelasan Kode Program :

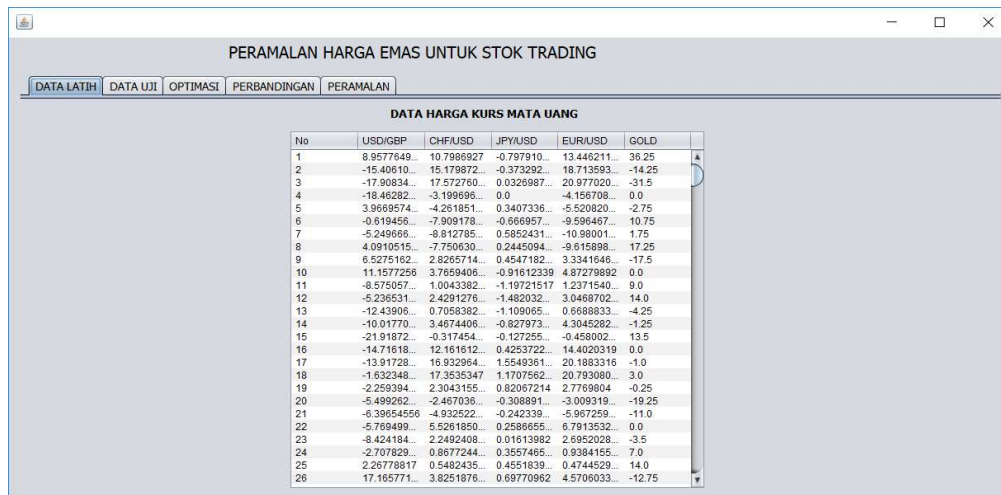
1. Baris 1 adalah deklarasi method proses defuzzifikasi.
2. Baris 2-16 adalah proses menghitung defuzzifikasi dengan menggunakan nilai alfa-predikat dan nilai Z. Pada proses ini menggunakan kondisi dimana bila salah satu nilai alfa-predikat atau Z bernilai 0, maka hasil perhitungannya adalah 0.
3. Baris 17-24 adalah proses menghitung nilai *error*, *error* kuadrat, dan total *error* kuadrat. Ketiga nilai ini kemudian disimpan kedalam kromosom untuk selanjutnya ditampilkan pada proses optimasi.
4. Baris 25-29 adalah proses mencari nilai RMSE dan nilai *fitness* yang akan digunakan sebagai nilai evaluasi. Nilai RMSE akan digunakan untuk mengevaluasi performa metode Tsukamoto, sedangkan nilai *fitness* akan digunakan pada evaluasi metode Algoritme Genetika.

5.3 Implementasi Antarmuka

Implementasi antarmuka ini adalah implementasi tampilan yang dapat digunakan oleh pengguna untuk melakukan optimasi maupun peramalan harga emas. Pada tampilan antarmuka ini akan ditampilkan beberapa laman yang ada pada sistem. Implementasi antarmuka yang ada meliputi laman Data, laman Perhitungan, laman Perbandingan, dan laman Proses Tsukamoto .

5.3.1 Implementasi Antarmuka Laman Data Latih

Laman data latih adalah laman yang berisi informasi tentang data kurs mata uang yang digunakan. Laman ini berisi kurs mata uang USD/GBP, CHF/USD, JPY/USD, EUR/USD dan harga emas. Implementasi antarmuka laman data latih terdapat pada Gambar 5.1.

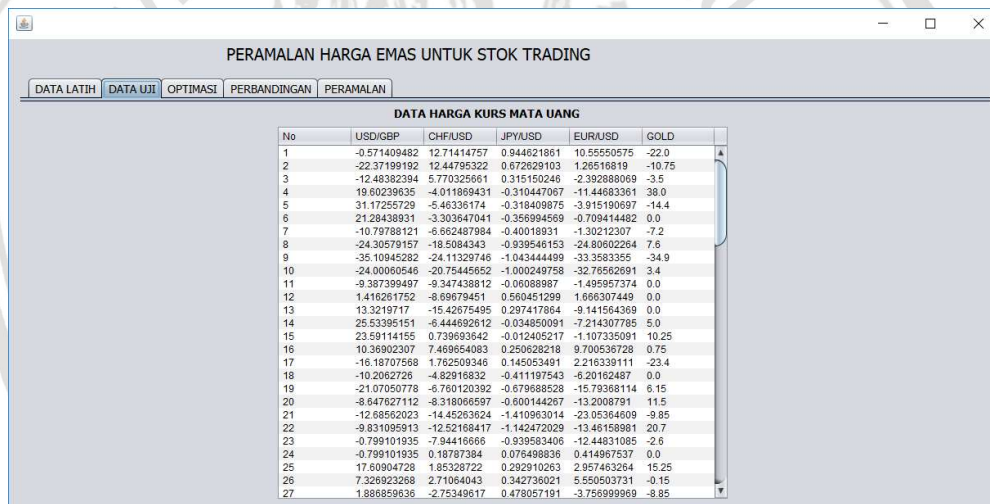


No	USD/GBP	CHF/USD	JPY/USD	EUR/USD	GOLD
1	8.9577649...	10.7986927...	-0.797910...	13.446211...	36.25
2	-15.40610...	15.179872...	-0.373292...	18.713593...	-14.25
3	-17.90934...	17.572760...	0.0328987...	20.977020...	-31.5
4	-18.46282...	-3.199696...	0.0	-4.156708...	0.0
5	3.969574...	-4.261851...	0.3407336...	-5.520820...	-2.75
6	-0.619456...	-7.909178...	-0.666957...	-9.596467...	10.75
7	-5.249666...	-8.812785...	0.5852431...	-10.98001...	1.75
8	4.0910515...	-7.750630...	0.2445094...	-9.615898...	17.25
9	6.5275162...	2.8265714...	0.4547182...	3.3341646...	-17.5
10	11.1577256...	3.7659406...	-0.91612339...	4.87279892...	0.0
11	-8.575057...	1.0043382...	-1.19721517...	1.2371540...	9.0
12	-5.236531...	2.4291276...	-1.482032...	3.0468702...	14.0
13	-12.43906...	0.7056382...	-1.109065...	0.6689833...	-4.25
14	-10.01770...	3.4674406...	-0.827973...	4.3045282...	-1.25
15	-21.91872...	-0.317454...	-0.127255...	-0.458002...	13.5
16	-14.71618...	12.161612...	0.4253722...	14.4020319...	0.0
17	-13.91728...	16.932964...	1.5549361...	20.1883316...	-1.0
18	-1.632348...	17.3535347...	1.1707562...	20.793060...	3.0
19	-2.259394...	2.3043155...	0.82067214...	2.7769804...	-0.25
20	-5.499262...	-2.467036...	-0.308891...	-3.009319...	-19.25
21	-6.39654556...	-4.932522...	-0.242339...	-5.967259...	-11.0
22	-5.769499...	5.5261850...	0.2586655...	6.7913532...	0.0
23	-8.424184...	2.2492408...	0.01613982...	2.6952028...	-3.5
24	-2.707829...	0.8677244...	0.3557465...	0.9384155...	7.0
25	2.26778817...	0.5482435...	0.4551839...	0.4744529...	14.0
26	17.165771...	3.8251876...	0.69770962...	4.5706033...	-12.75

Gambar 5.1 Antarmuka Laman Data Latih

5.3.2 Implementasi Anatrmuka Laman Data Uji

Laman data uji adalah laman yang berisi informasi tentang data kurs mata uang yang digunakan untuk peramalan harga emas dengan menggunakan metode FIS Tsukamoto. Laman ini berisi kurs mata uang USD/GBP, CHF/USD, JPY/USD, EUR/USD dan harga emas. Implementasi antarmuka laman data uji terdapat pada Gambar 5.2.



No	USD/GBP	CHF/USD	JPY/USD	EUR/USD	GOLD
1	-0.57140482	12.71414757	0.944621861	10.5550575	-22.0
2	-22.37199192	12.44795322	0.672629103	1.26519819	-10.75
3	-12.46382394	5.77032561	0.315150249	-2.392888069	-3.5
4	19.60239635	-4.011869431	-0.310447067	-11.44683361	38.0
5	31.17255729	-5.46336174	-0.318409875	-3.915190697	-14.4
6	21.28438931	-3.303647041	-0.356994569	-0.709414482	0.0
7	-10.79788121	-6.662487984	-0.40018931	-1.30212307	-7.2
8	-24.30579157	-18.5084343	-0.939546153	-24.80602264	7.5
9	-35.10945282	-24.11329746	-1.043444499	-33.3583355	-34.9
10	-24.00060546	-20.75445652	-1.000249758	-32.78562691	3.4
11	-9.387399497	-9.347438812	-0.06088987	-1.495957374	0.0
12	1.416261752	-8.69679451	0.560451299	1.666307449	0.0
13	13.3219717	-15.42675495	0.297417864	-9.141564369	0.0
14	25.53395151	-6.444692612	-0.034850091	-7.214307785	5.0
15	23.59114155	0.739693642	-0.012405217	-1.107335091	10.25
16	10.36902307	7.469654083	0.250628218	9.700536728	0.75
17	-16.18707568	1.762509346	0.145053491	2.216339111	-23.4
18	-10.2062726	-4.82916832	-0.411197543	-6.20162487	0.0
19	-21.07050778	-6.760120392	-0.679688528	-15.79368114	6.15
20	-8.647627112	-8.318066597	-0.600144267	-13.2008791	11.5
21	-12.68562023	-14.45263924	-1.410963014	-23.05364609	-9.85
22	-9.831085913	-12.52168417	-1.142472029	-13.46158981	20.7
23	-0.799101935	-7.94416666	-0.939583406	-12.44831085	-2.6
24	-0.799101935	0.18787384	0.076498836	0.414967537	0.0
25	17.60904728	1.85328722	0.292910263	2.957463264	15.25
26	7.32692368	2.71064043	0.342736021	5.55050731	-0.15
27	1.886859636	-2.75349617	0.478057191	-3.756999959	-8.85

Gambar 5.2 Antarmuka Laman Data Uji

5.3.3 Implementasi Antarmuka Laman Optimasi

Laman Optimasi berisikan perhitungan optimasi dengan Algoritme Genetika. Laman ini memuat *textbox* untuk mengisi masukan yang dibutuhkan, tombol untuk melakukan proses, *field* yang berisikan informasi tentang hasil perhitungan, dan *field* yang berisi hasil rekomendasi fungsi keanggotaan dengan nilai *fitness* terbaik. Implementasi laman Perhitungan dapat dilihat pada Gambar 5.3.

PERAMALAN HARGA EMAS UNTUK STOK TRADING

DATA LATIH DATA UJI **OPTIMASI** PERBANDINGAN PERAMALAN

Input Parameter Algoritma Genetika

Crossover Rate: Populasi:
Mutation Rate: Iterasi:

PROSES

Populasi Iterasi Terakhir

Parent	g1	g2	g3	g4	g5	g6	g7	g8	g9	g10	g11	g12	g13	g14	g15	g16	g17	g18	g19	g20	Fitness
P1	6.5	9.3	29.9	38.0	15.0	21.8	25.6	26.5	1.4	1.7	2.8	3.6	0.3	1.2	26.7	35.0	7.0	12.2	17.3	29.4	7.491...
P2	6.5	9.3	29.9	38.0	15.0	21.8	25.6	26.5	1.4	1.7	2.8	3.6	0.3	1.2	26.7	35.0	7.0	12.2	17.3	29.4	7.491...
P3	6.5	9.3	29.9	38.0	15.0	21.8	25.6	26.5	1.4	1.7	2.8	3.6	0.3	1.2	26.7	35.0	7.0	12.2	17.3	29.4	7.491...
P4	6.5	9.3	29.9	38.0	15.0	21.8	25.6	26.5	1.4	1.7	2.8	3.6	0.3	1.2	26.7	35.0	7.0	12.2	17.3	29.4	7.491...
P5	6.5	9.3	29.9	38.0	15.0	21.8	25.6	26.5	1.4	1.7	2.8	3.6	0.3	1.2	26.7	35.0	7.0	12.2	17.3	29.4	7.491...
P6	6.5	9.3	29.9	38.0	15.0	21.8	25.6	26.5	1.4	1.7	2.8	3.6	0.3	1.2	26.7	35.0	7.0	12.2	17.3	29.4	7.491...
P7	6.5	9.3	29.9	38.0	15.0	21.8	25.6	26.5	1.4	1.7	2.8	3.6	0.3	1.2	26.7	35.0	7.0	12.2	17.3	29.4	7.491...
P8	6.5	9.3	29.9	38.0	15.0	21.8	25.6	26.5	1.4	1.7	2.8	3.6	0.3	1.2	26.7	35.0	7.0	12.2	17.3	29.4	7.491...
P9	6.5	9.3	29.9	38.0	15.0	21.8	25.6	26.5	1.4	1.7	2.8	3.6	0.3	1.2	26.7	35.0	7.0	12.2	17.3	29.4	7.491...
P10	6.5	9.3	29.9	38.0	15.0	21.8	25.6	26.5	1.4	1.7	2.8	3.6	0.3	1.2	26.7	35.0	7.0	12.2	17.3	29.4	7.491...
P11	6.5	9.3	29.9	38.0	15.0	21.8	25.6	26.5	1.4	1.7	2.8	3.6	0.3	1.2	26.7	35.0	7.0	12.2	17.3	29.4	7.491...
P12	6.5	9.3	29.9	38.0	15.0	21.8	25.6	26.5	1.4	1.7	2.8	3.6	0.3	1.2	26.7	35.0	7.0	12.2	17.3	29.4	7.491...
P13	6.5	9.3	29.9	38.0	15.0	21.8	25.6	26.5	1.4	1.7	2.8	3.6	0.3	1.2	26.7	35.0	7.0	12.2	17.3	29.4	7.491...
P14	6.5	9.3	29.9	38.0	15.0	21.8	25.6	26.5	1.4	1.7	2.8	3.6	0.3	1.2	26.7	35.0	7.0	12.2	17.3	29.4	7.491...

Hasil Rekomendasi

Parent	g1	g2	g3	g4	g5	g6	g7	g8	g9	g10	g11	g12	g13	g14	g15	g16	g17	g18	g19	g20	Fitness
P1	6.5	9.3	29.9	38.0	15.0	21.8	25.6	26.5	1.4	1.7	2.8	3.6	0.3	1.2	26.7	35.0	7.0	12.2	17.3	29.4	7.491...

Gambar 5.3 Implementasi Laman Optimasi

5.3.4 Implementasi Antarmuka Laman Perbandingan

Laman perbandingan memuat total *error*, nilai RMSE, dan nilai fitness tertinggi setiap iterasi secara lengkap. Pada laman ini menuliskan nilai semua hasil perhitungan tiap iterasi secara lengkap. Implementasi antarmuka laman perbandingan dapat dilihat pada Gambar 5.4.

PERAMALAN HARGA EMAS UNTUK STOK TRADING

DATA LATIH DATA UJI **PERBANDINGAN** PERAMALAN

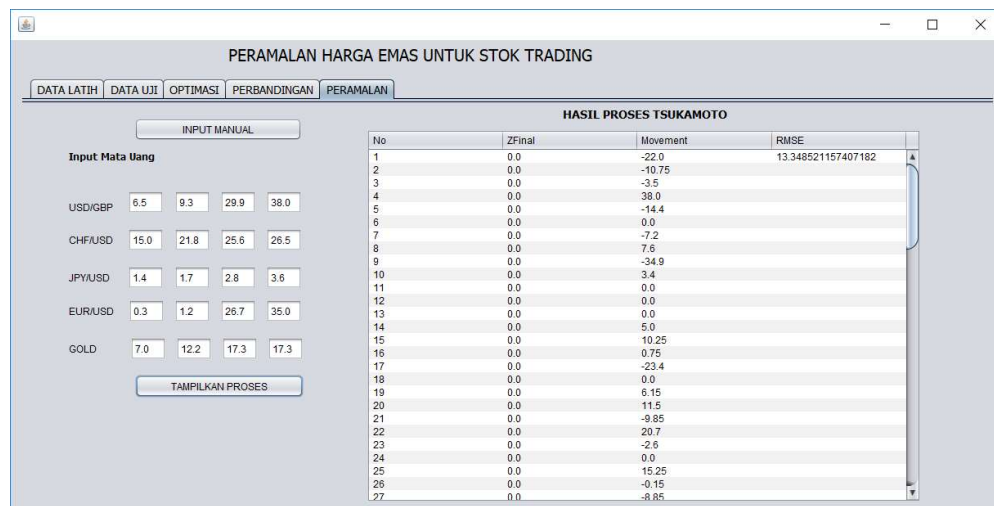
HASIL SETIAP ITERASI ALGORITMA GENETIKA

Iterasi	Total Error	RMSE	Fitness
1	19737.843310982837	14.049143500933727	7.117871633480991
2	18984.852577921514	13.778371696463935	7.257751672020609
3	18600.85216437445	13.638484112025143	7.332187789640331
4	18084.74034388863	13.44795164472591	7.436076708322976
5	17997.28548819652	13.41539618803579	7.454122010141055
6	17831.125285478036	13.35332366322259	7.4887722728849635
7	17831.125285478036	13.35332366322259	7.4887722728849635
8	17831.125285478036	13.35332366322259	7.4887722728849635
9	17831.125285478036	13.35332366322259	7.4887722728849635
10	17831.125285478036	13.35332366322259	7.4887722728849635
11	17831.125285478036	13.35332366322259	7.4887722728849635
12	17818.30170897472	13.348521157407182	7.4914665692768025
13	17818.30170897472	13.348521157407182	7.4914665692768025
14	17818.30170897472	13.348521157407182	7.4914665692768025
15	17818.30170897472	13.348521157407182	7.4914665692768025
16	17818.30170897472	13.348521157407182	7.4914665692768025
17	17818.30170897472	13.348521157407182	7.4914665692768025
18	17818.30170897472	13.348521157407182	7.4914665692768025
19	17818.30170897472	13.348521157407182	7.4914665692768025
20	17818.30170897472	13.348521157407182	7.4914665692768025
21	17818.30170897472	13.348521157407182	7.4914665692768025
22	17818.30170897472	13.348521157407182	7.4914665692768025
23	17818.30170897472	13.348521157407182	7.4914665692768025

Gambar 5.4 Implementasi Laman Perbandingan

5.3.5 Implementasi Antarmuka Laman Peramalan

Laman Peramalan berisikan indeks harga peramalan dengan menggunakan metode Tsukamoto. Laman ini berisikan indeks harga peramalan, harga asli, dan RMSE. Di laman ini juga menyediakan sebuah kolom kosong apabila *user* ingin memasukkan data fungsi keanggotaan secara manual kedalam sistem. Implementasi antarmuka laman Proses Tsukamoto dapat dilihat pada Gambar 5.5.



Gambar 5.5 Implementasi Laman Peramalan



BAB 6 PENGUJIAN DAN ANALISIS

Pada bab 6 pengujian ini akan menjelaskan tentang pengujian proses yang telah dibuat sebelumnya, parameter-parameter yang ada akan sangat memengaruhi hasil solusi. Parameter-parameter yang terbaik ini akan memberikan himpunan solusi yang baik pula pada pengujian ini.

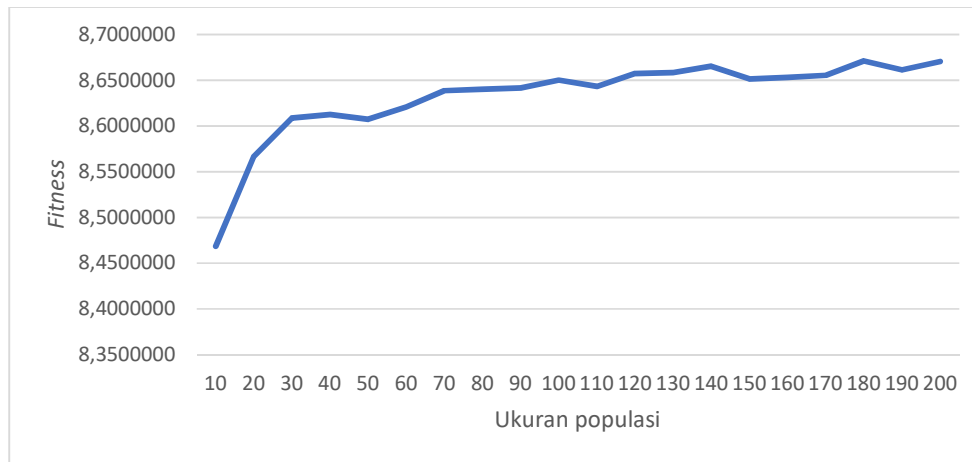
6.1 Pengujian Ukuran Populasi dan Analisis

Pengujian ini bertujuan untuk mencari ukuran populasi terbaik, hasil dari pengujian ini akan digunakan sebagai parameter ukuran populasi pada pengujian selanjutnya agar didapatkan hasil yang optimal. Ukuran populasi yang digunakan pada pengujian ini merupakan kelipatan 10 dimulai dari 10 sampai dengan 200. Parameter lain yang ditetapkan antara lain nilai $cr = 0.5$, $mr = 0.5$, dan maksimal iterasi = 1000. Setiap kondisi akan dilakukan 10 kali proses percobaan, lalu diambil nilai rata-ratanya untuk diambil nilai tertinggi sebagai hasil ukuran populasi yang terbaik. Berikut ini adalah tabel pengujian ukuran populasi yang ditunjukkan pada Tabel 6.1. Untuk lebih lengkapnya dapat dilihat pada Lampiran B.

Tabel 6.1 Pengujian Populasi

Populasi	Percobaan ke -							Rata-Rata
	1	2	3	4	5	...	10	
10	8,6325	8,3511	8,6101	8,5881	8,3922	...	8,1149	8,4687
20	8,6361	8,5481	8,6069	8,5139	8,5614	...	8,6256	8,5666
30	8,5606	8,5741	8,6491	8,6258	8,6184	...	8,6121	8,6089
40	8,6438	8,5776	8,5991	8,6230	8,6338	...	8,6233	8,6126
50	8,6184	8,6165	8,5767	8,5988	8,6230	...	8,6278	8,6075
60	8,6140	8,6110	8,6270	8,6261	8,6233	...	8,6351	8,6210
70	8,6760	8,6238	8,6390	8,6278	8,6535	...	8,6502	8,6386
80	8,6378	8,6282	8,6754	8,6803	8,6109	...	8,6590	8,6402
90	8,6373	8,6501	8,6164	8,6537	8,6238	...	8,6027	8,6417
100	8,6328	8,6458	8,6450	8,6487	8,6925	...	8,6508	8,6502
...
200	8,6738	8,6640	8,6995	8,6541	8,6601	...	8,6537	8,6707

Pada Tabel 6.1 ditunjukkan bahwa hasil pengujian ukuran populasi menghasilkan nilai *fitness* yang berbeda dari tiap masukan ukuran populasi. Ukuran populasi terbaik didapatkan pada masukan 180 populasi, perbedaan masukan ukuran populasi menghasilkan nilai *fitness* yang berbeda pula. Agar lebih mudah melihat perbedaan nilai serta menganalisa hasil pengujian yang ada, maka hasil pengujian tabel diatas akan digambarkan dalam bentuk grafik seperti pada Gambar 6.1.



Gambar 6.1 Pengujian Ukuran Populasi

Pada Gambar 6.1 dapat dilihat bahwa masukan ukuran populasi sangat berpengaruh pada nilai *fitness* dari setiap individu. Pada perancangan awalnya hanya mengujikan populasi dengan rentang 0-100. Namun pada saat dilakukan proses pengujian, ukuran populasi 100 masih memiliki nilai *fitness* tertinggi sehingga pengujian pun dilanjutkan sampai ukuran populasi 200. Ukuran populasi terbaik didapatkan pada populasi sebesar 180, dengan rata-rata nilai *fitness* sebesar 8,6712. Sedangkan nilai *fitness* paling kecil didapatkan pada ukuran populasi sejumlah 10, dengan rata-rata nilai *fitness* sebesar 8,4686. Dengan demikian, dapat dikatakan bahwa ukuran populasi sangat mempengaruhi nilai *fitness*. Semakin besar ukuran populasi, maka akan memberikan nilai *fitness* yang baik pula. Dari grafik pengujian diatas terlihat bahwa terjadi peningkatan nilai *fitness*. Hal ini disebabkan karena ukuran populasi yang semakin besar akan memberikan ruang pencarian yang lebih luas, selain itu populasi yang besar akan menghasilkan keragaman individu yang lebih banyak ketika proses inisialisasi kromosom. Sehingga Algoritme Genetika dapat memaksimalkan pencarian himpunan solusi yang paling baik.

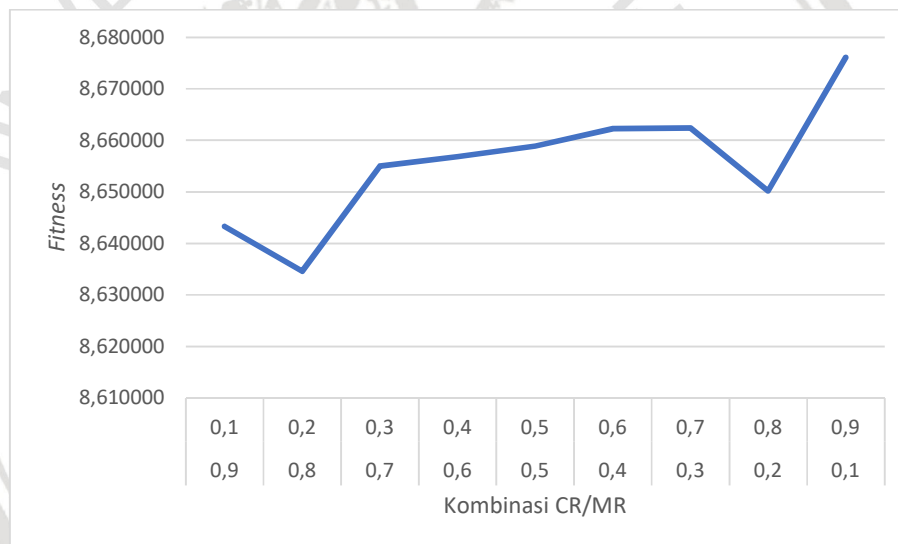
6.2 Pengujian Kombinasi *cr* dan *mr* dan Analisis

Pengujian ini memiliki tujuan untuk mencari kombinasi *cr* dan *mr* terbaik, hasil pengujian ini akan digunakan sebagai parameter kombinasi *cr* dan *mr* pada pengujian selanjutnya agar didapatkan hasil yang optimal. Kombinasi *cr* dan *mr* yang digunakan pada pengujian ini merupakan kelipatan 0,1 antara 0 dan 1. Nilai *cr* yang digunakan dimulai dari 0,1-0,9, sedangkan nilai *mr* yang digunakan dimulai dari 0,9-0,1. Untuk parameter ukuran populasi terbaik didapatkan dari pengujian ukuran populasi sebelumnya seperti pada Subbab 6.1, dan maksimal generasi = 1000. Setiap kondisi akan dilakukan 10 kali proses percobaan, lalu diambil nilai rata-ratanya untuk kemudian diambil nilai tertinggi sebagai kombinasi *cr* dan *mr* terbaik. Berikut ini adalah tabel pengujian kombinasi *cr* dan *mr* yang ditunjukkan pada Tabel 6.2. Untuk lebih lengkapnya dapat dilihat pada Lampiran B.

Tabel 6.2 Pengujian Kombinasi *cr* dan *mr*

Kombinasi		Percobaan ke -						Rata-Rata
<i>cr</i>	<i>mr</i>	1	2	3	4	10	
0.1	0.9	8,6293	8,6516	8,6800	8,6872	...	8,6412	8,6433
0.2	0.8	8,6274	8,6372	8,6178	8,6483	...	8,6267	8,6346
0.3	0.7	8,6749	8,6547	8,6600	8,6557	...	8,6613	8,6550
0.4	0.6	8,6459	8,6664	8,6179	8,6385	...	8,6680	8,6568
0.5	0.5	8,6277	8,6825	8,6726	8,6440	...	8,7009	8,6589
0.6	0.4	8,6726	8,6701	8,6800	8,6517	...	8,6216	8,6623
0.7	0.3	8,6358	8,6495	8,6797	8,6893	...	8,6735	8,6624
0.8	0.2	8,6475	8,6298	8,6763	8,6598	...	8,6575	8,6502
0.9	0.1	8,6869	8,6560	8,6918	8,6694	...	8,6781	8,6761

Pada Tabel 6.2 hasil pengujian kombinasi *cr* dan *mr* menghasilkan nilai *fitness* yang berbeda pada setiap masukan kombinasi *cr* dan *mr*. Pada pengujian kombinasi *cr* dan *mr* ini parameter ukuran populasi yang digunakan adalah 180, yang mana nilai tersebut diperoleh dari pengujian ukuran populasi yang sebelumnya telah dilakukan. Agar lebih mudah melihat perbedaan nilai serta menganalisa hasil pengujian yang ada, maka hasil pengujian tabel diatas akan digambarkan dalam bentuk grafik seperti pada Gambar 6.2.



Gambar 6.2 Pengujian Kombinasi *cr* dan *mr*

Pada Gambar 6.2 dapat dilihat bahwa kombinasi *cr* dan *mr* terbaik yang didapatkan adalah *cr*=0,9 dan *mr*=0,1, dengan rata-rata nilai *fitness* sebesar 8,6761. pengujian ini menggunakan parameter ukuran populasi terbaik yang didapatkan dari pengujian sebelumnya, yaitu 180 populasi. Sedangkan untuk nilai *fitness* terkecil didapatkan *cr*=0,1 dan *mr*=0,9, dengan rata-rata nilai *fitness* sebesar 8,6433. Pada pengujian ini nilai *cr* terbaik yang didapatkan sangat tinggi, sebaliknya nilai *mr* yang didapatkan terlalu rendah. Pada penelitian yang menggunakan metode Algoritme Genetika sebagai solusi permasalahan sebaiknya

nilai *crossover rate* cukup tinggi dan nilai *mutation rate* sebaiknya menggunakan nilai yang lebih rendah (Desiani & Arhami, 2006).

Alasannya nilai *crossover rate* sebaiknya cukup tinggi dikarenakan tujuan dari *crossover* adalah mendapatkan kromosom-kromosom baru yang dapat dijadikan solusi pada generasi selanjutnya dengan *fitness* yang lebih tinggi untuk memperoleh solusi paling optimal. *Crossover rate* akan mengontrol kemampuan Algoritme Genetika dalam mengeksplorasi pencarian yang ada untuk mencapai lokal optimum. Namun, semakin tinggi *crossover rate* maka akan semakin cepat pula proses eksploitasi ini berlangsung sehingga akan menyebabkan konvergensi dini (Lin, et al., 2003). Maka dibutuhkan *mutation rate* untuk mengontrol kecepatan Algoritme Genetika dalam menjelajahi area baru. Operasi mutasi dapat digunakan untuk menghindari konvergensi dini dan menjaga diversitas individu pada populasi. Karena proses mutasi dilakukan secara acak untuk mengubah nilai gen yang ada pada kromosom maka *offspring* hasil proses mutasi ini tidak dijamin memiliki *fitness* yang lebih baik karena dianggap dapat mengganggu kualitas kromosom yang telah didapatkan sebelumnya. Namun, mutasi tetap dilakukan karena proses ini memiliki kemungkinan untuk menghasilkan *offspring* dengan nilai *fitness* yang lebih baik dari sebelumnya meskipun probabilitasnya cukup rendah (Desiani & Arhami, 2006). Berdasarkan penelitian yang dilakukan oleh Fogarty (1989), ketika nilai *mutation rate* diturunkan secara eksponensial pada setiap generasi maka akan memberikan performa yang lebih baik.

Kombinasi *cr* dan *mr* sangat mempengaruhi perilaku dan kinerja dari Algoritme Genetika. Umumnya nilai *cr* berada di kisaran 0,5-1,0, sedangkan nilai *mr* berada di kisaran 0,001-0,05. Panduan umum ini diambil dari studi-studi empiris pada serangkaian masalah tes tetap. Namun, hal ini tidak selalu bisa dijadikan patokan karena penggunaan optimal *cr* dan *mr* akan spesifik tergantung masalah yang ada (Lin, et al., 2003). Setiap permasalahan memiliki karakter masing-masing, sehingga kombinasi *cr* dan *mr* terbaik yang didapatkan pada penelitian ini belum tentu cocok dengan permasalahan lain meskipun menggunakan metode yang sama.

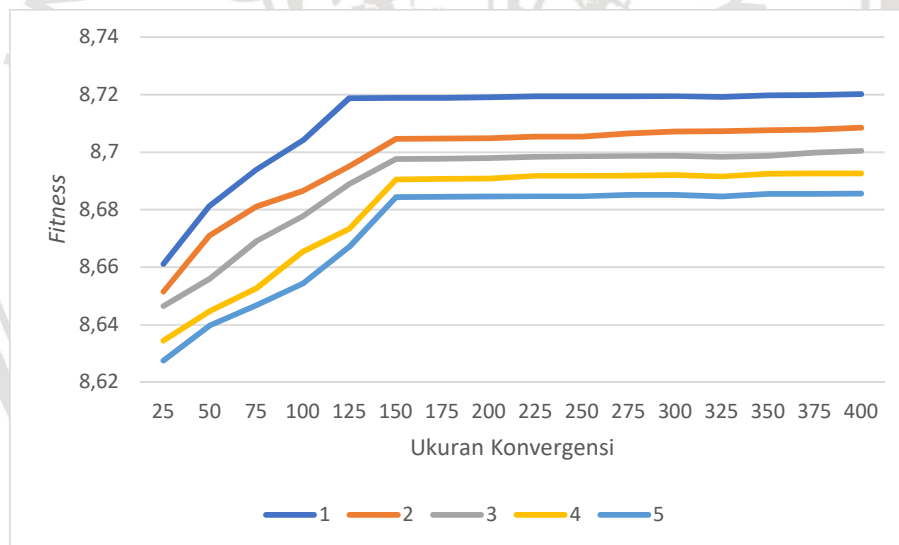
6.3 Pengujian Konvergensi dan Analisis

Pengujian konvergensi ini bertujuan untuk mencari jumlah generasi yang terbaik serta menggabungkan seluruh parameter terbaik sehingga didapatkan solusi akhir jumlah produksi yang optimal. Jumlah generasi yang digunakan pada pengujian ini merupakan kelipatan 25 dimulai dari 25 sampai dengan 400. Untuk parameter ukuran populasi, digunakan nilai ukuran populasi yang didapatkan dari pengujian ukuran populasi pada Subbab 6.1. Untuk parameter kombinasi *cr* dan *mr*, digunakan nilai kombinasi *cr* dan *mr* yang didapatkan dari pengujian kombinasi *cr* dan *mr* pada Subbab 6.2. Setiap kondisi akan dicobakan sebanyak 10 kali, kemudian akan diambil sebuah nilai dengan rata-rata paling tinggi untuk dijadikan jumlah generasi yang terbaik. Hasil terbaik dari pengujian ini adalah hasil solusi akhir dari proses optimasi, hasil tersebut adalah keluaran fungsi keanggotaan setiap variabel yang ada. Berikut adalah tabel pengujian konvergensi yang ditunjukkan pada Tabel 6.3. Untuk lebih lengkapnya dapat dilihat pada Lampiran B.

Tabel 6.3 Pengujian Konvergensi

Iterasi	Percobaan ke -							Rata-Rata
	1	2	3	4	5	...	10	
25	8,6344	8,6466	8,6612	8,6190	8,6153	...	8,6276	8,6325
50	8,6447	8,6560	8,6814	8,6274	8,6261	...	8,6398	8,6435
75	8,6527	8,6691	8,6940	8,6281	8,6274	...	8,6468	8,6503
100	8,6654	8,6777	8,7041	8,6292	8,6287	...	8,6544	8,6572
125	8,6734	8,6889	8,7187	8,6395	8,6394	...	8,6671	8,6673
150	8,6905	8,6976	8,7189	8,6610	8,6477	...	8,6844	8,6827
175	8,6907	8,6978	8,7189	8,6610	8,6492	...	8,6845	8,6833
200	8,6908	8,6979	8,7190	8,6632	8,6544	...	8,6845	8,6844
225	8,6918	8,6984	8,7193	8,6670	8,6591	...	8,6846	8,6857
250	8,6918	8,6985	8,7194	8,6675	8,6600	...	8,6847	8,6860
...
400	8,6926	8,6995	8,7196	8,6666	8,6582	...	8,6857	8,6864

Berdasarkan Tabel 6.3 hasil pengujian konvergensi ini merupakan gabungan parameter terbaik dari ukuran populasi serta kombinasi cr dan mr . Setiap pengujian jumlah generasi ini memberikan nilai *fitness* yang berbeda-beda. Untuk lebih mempermudah melihat perbedaannya serta mengkaji hasil pengujian nya, isi dari tabel diatas akan disajikan dalam bentuk grafik seperti pada Gambar 6.3.



Gambar 6.3 Pengujian Konvergensi

Pada Gambar 6.3 menampilkan 5 *fitness* tertinggi dari setiap iterasi. Dari Gambar 6.3 dapat dilihat bahwa setelah menggabungkan parameter terbaik dari pengujian yang telah dilakukan sebelumnya. Jumlah generasi terbaik yang diperoleh adalah sebesar 325 iterasi dengan nilai *fitness* rata-rata sebesar 8,6870 menggunakan populasi sejumlah 180 serta nilai $cr=0,9$ dan $mr=0,1$. Sedangkan nilai *fitness* tertinggi terdapat di iterasi ke 325 dengan nilai *fitness* 8,7194. Pada

pengujian ini terlihat bahwa di setiap iterasi nilai *fitness* akan terus mengalami kenaikan, kemudian setelah itu pergerakan *fitness* akan stabil. Hal ini terjadi karena semakin banyak iterasi maka Algoritme Genetika memiliki ruang pencarian yang lebih besar. Sehingga dengan banyaknya iterasi memiliki kemungkinan untuk mendapatkan solusi yang terbaik, meski tidak selalu iterasi terbesar akan memberikan solusi terbaik.

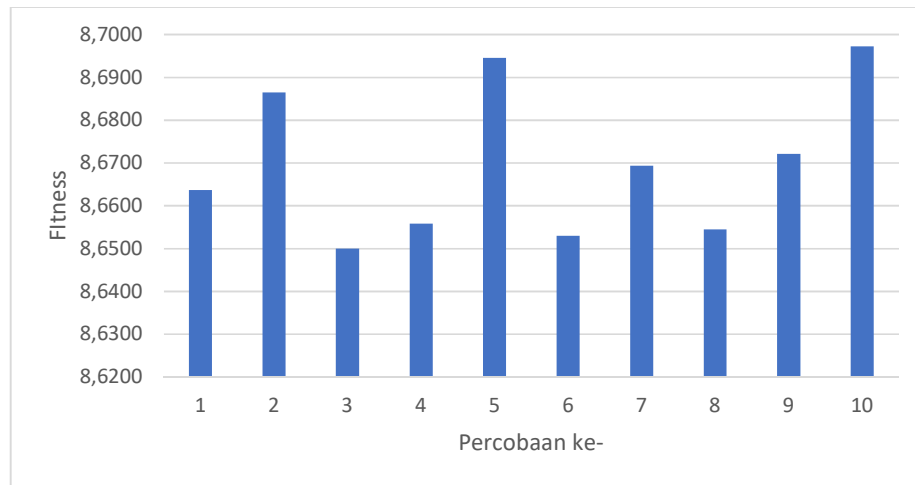
6.4 Pengujian Dengan Parameter Terbaik

Pengujian dengan parameter terbaik ini bertujuan untuk mencari individu terbaik berdasarkan parameter terbaik yang telah didapatkan sebelumnya. Untuk parameter ukuran populasi, digunakan nilai ukuran populasi yang didapatkan dari pengujian ukuran populasi pada Subbab 6.1. Untuk parameter kombinasi *cr* dan *mr*, digunakan nilai kombinasi *cr* dan *mr* yang didapatkan dari pengujian kombinasi *cr* dan *mr* pada Subbab 6.2. Dan parameter iterasi didapatkan dari pengujian konvergensi pada Subbab 6.3. Kondisi ini akan dicobakan sebanyak 10 kali, kemudian akan diambil sebuah nilai dengan rata-rata paling tinggi untuk dijadikan individu terbaik. Berikut adalah tabel pengujian dengan parameter terbaik yang ditunjukkan pada Tabel 6.4.

Tabel 6.4 Pengujian Parameter Terbaik

Percobaan ke-	Populasi	Crossover Rate	Mutation Rate	Konvergensi	Fitnesss
1	180	0,9	0,1	325	8,6637
2					8,6866
3					8,6500
4					8,6558
5					8,6946
6					8,6530
7					8,6694
8					8,6545
9					8,6721
10					8,6973

Hasil gabungan dari parameter terbaik dengan ukuran populasi = 180, kombinasi *cr* = 0,9 dan *mr*=0,1, serta ukuran konvergensi = 325 ini memberikan hasil solusi dengan nilai *fitness* yang berada antara 8,6500 hingga 8,6973. Untuk lebih jelasnya dapat dilihat pada Gambar 6.4.



Gambar 6.4 Pengujian dengan Parameter Terbaik

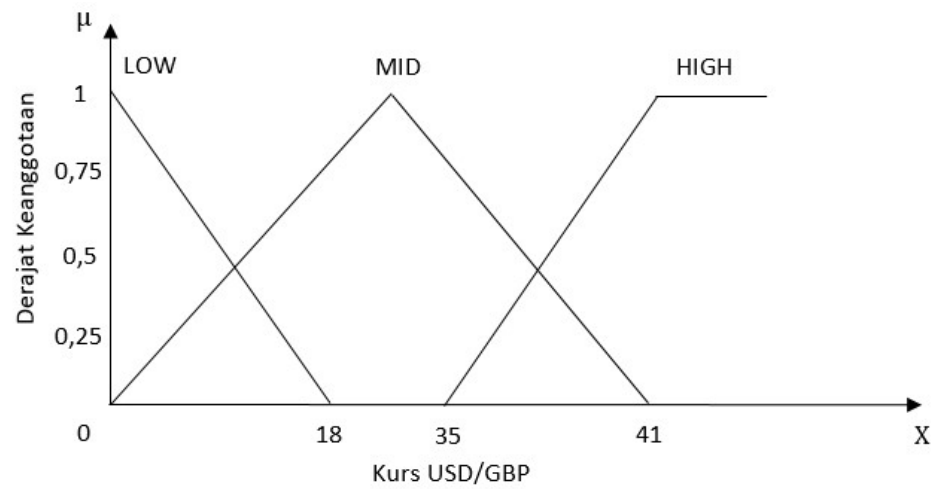
Karena pengujian ini adalah pengujian gabungan dari pengujian dengan semua parameter terbaik yang telah diperoleh, maka dari hasil pengujian ini juga akan diambil hasil solusi dari permasalahan optimasi fungsi keanggotaan FIS Tsukamoto. Untuk mendapatkan hasil solusinya, maka akan diambil individu dengan *fitness* terbaik dari 10 percobaan yang dilakukan. Individu yang memiliki *fitness* terbaik dari percobaan ini adalah individu percobaan ke-10 dengan nilai *fitness* sebesar 8,6973. Agar lebih mudah, gen individu yang terpilih tersebut dapat dilihat pada Tabel 6.5. Dan penggambaran fungsi keanggotaannya dapat dilihat pada Gambar 6.5 sampai Gambar 6.9.

Tabel 6.5 Individu Terbaik

USD/GBP				CHF/USD				JPY/USD			
0	18	35	41	4,4	4,7	14	33,2	1,36	1,5	2,7	2,81

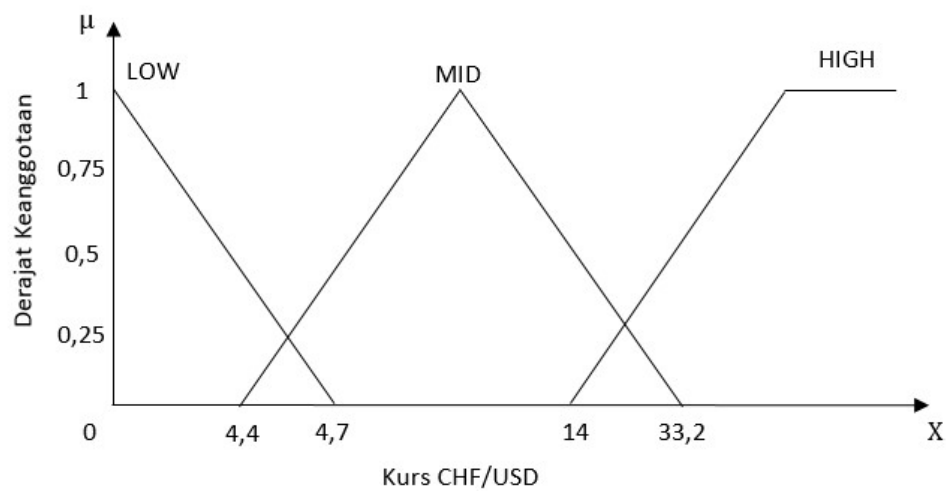
EUR/USD				GOLD			
2	4	10	23,2	2,5	11	12	30,1

- USD/GBP

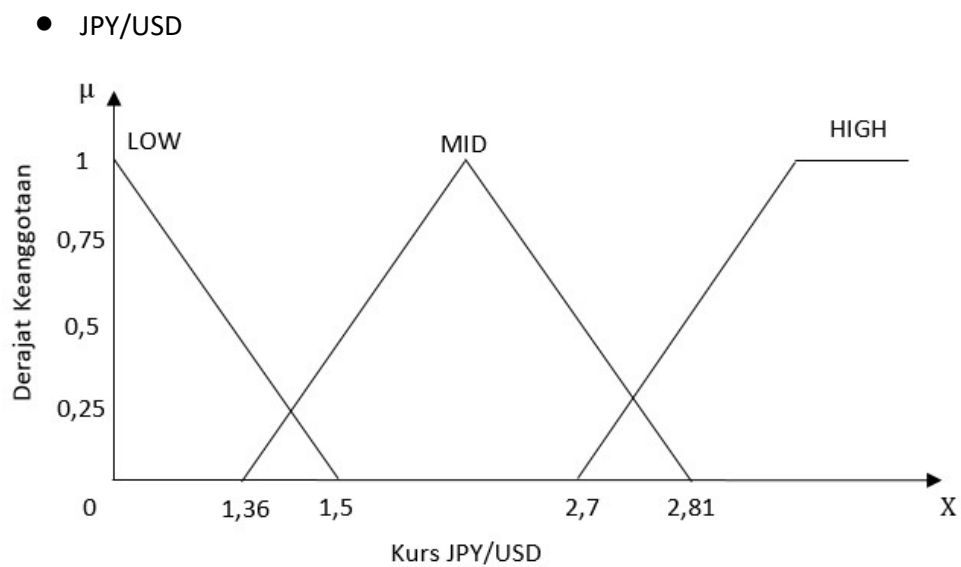


Gambar 6.5 Fungsi Keanggotaan USD/GBP

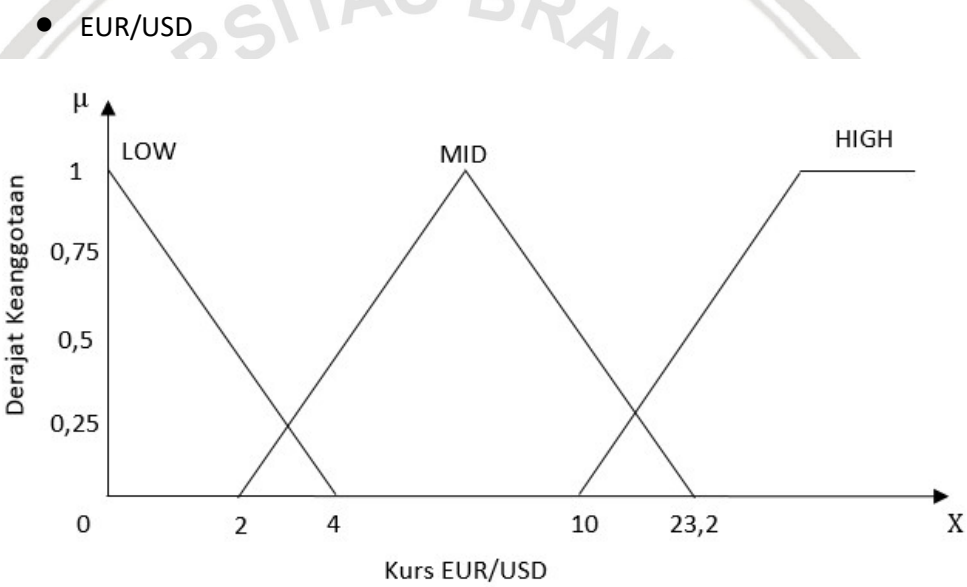
- CHF/USD



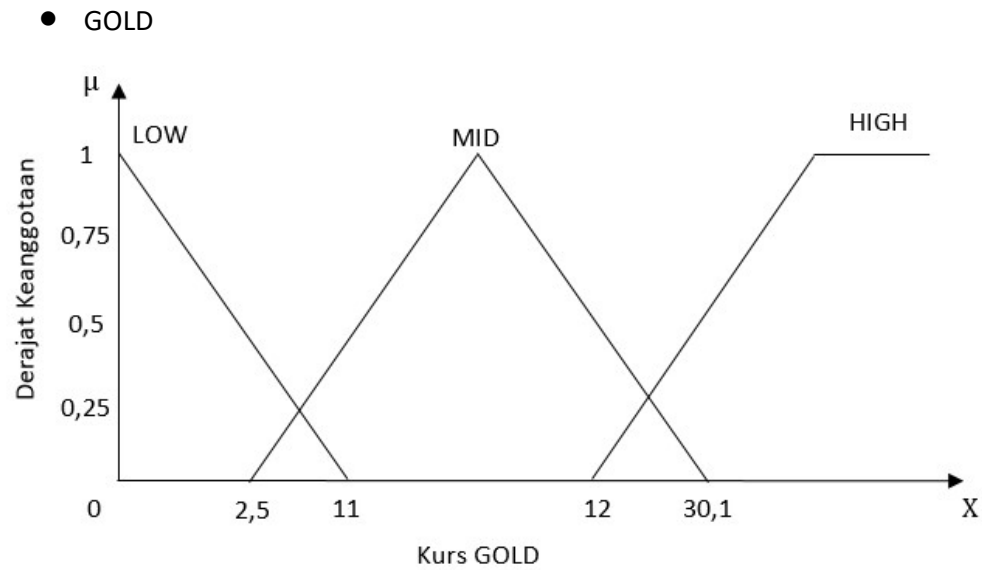
Gambar 6.6 Fungsi Keanggotaan CHF/USD



Gambar 6.7 Fungsi Keanggotaan JPY/USD



Gambar 6.8 Fungsi Keanggotaan EUR/USD



Gambar 6.9 Fungsi Keanggotaan GOLD

6.5 Analisis Global

Setelah fungsi keanggotaan hasil optimasi didapatkan selanjutnya fungsi keanggotaan ini akan digunakan untuk meramalkan harga emas menggunakan metode FIS Tsukamoto menggunakan 100 data uji. Untuk mendapatkan nilai *error* dari prediksi ini, akan digunakan Pengujian RMSE. Pengujian RMSE ini bertujuan untuk mencari nilai *error* yang dihasilkan oleh sistem peramalan dengan Tsukamoto. RMSE akan memberikan hasil yang relatif terhadap nilai yang sebenarnya sehingga nilai kesalahan yang didapatkan menjadi lebih stabil. Pengujian peramalan harga akan dilakukan dengan RMSE dan dihitung menggunakan Persamaan 2.5. Individu terbaik yang didapatkan dari pengujian parameter yang telah dilakukan sebelumnya kemudian akan dicobakan 10 kali dengan menggunakan data uji. Untuk lebih lengkapnya dapat dilihat pada Tabel 6.6.

Tabel 6.6 Perbandingan FIS dan GA-FIS

Percobaan ke-	FIS	GA-FIS
1	13,3611	12,6222
2		12,6206
3		12,6017
4		12,6215
5		12,6208
6		12,5936
7		12,5866
8		12,5801
9		12,6050
10		12,6009
Rata-Rata	13,3611	12,6052

Rata-rata nilai *error* yang dihasilkan oleh GA-FIS adalah sebesar 12,6052. Sedangkan nilai *error* yang dihasilkan oleh FIS saja adalah sebesar 13,3611. Nilai *error* terkecil yang dihasilkan oleh GA-FIS adalah sebesar 12,5801. Nilai *error* yang lebih rendah ini dihasilkan karena GA-FIS mampu membangkitkan fungsi keanggotaan berdasarkan data-data sebelumnya, sehingga fungsi keanggotaan ini dapat beradaptasi dengan data dan tidak bergantung kepada pakar. Memang selisih nilai *error*-nya tidak begitu jauh, namun dalam kasus peramalan tentu nilai *error* sangat berpengaruh dan tentu saja diharapkan sistem dapat menghasilkan nilai *error* sekecil mungkin.



BAB 7 PENUTUP

7.1 Kesimpulan

Berdasarkan hasil penelitian pengoptimalan fungsi keanggotaan FIS Tsukamoto menggunakan Algoritme Genetika pada peramalan harga emas untuk *stock trading* secara menyeluruh serta merujuk kepada permasalahan yang ada, maka dapat disimpulkan bahwa:

1. Fungsi keanggotaan FIS Tsukamoto akan direpresentasikan sebagai kromosom pada Algoritme Genetika. Setiap fungsi keanggotaan ini kemudian dilakukan peramalan dengan menggunakan FIS Tsukamoto dan dicari nilai RMSE-nya. Nilai RMSE ini kemudian akan digunakan pada Algoritme Genetika sebagai nilai *fitness*. Terdapat 4 parameter yang digunakan pada penelitian ini untuk mendapatkan hasil yang mendekati optimal. Tiga parameter untuk Algoritme Genetika antara lain ukuran populasi, kombinasi *cr* dan *mr* dan jumlah generasi. Sedangkan satu parameter lain digunakan pada FIS Tsukamoto yaitu nilai RMSE. Pengujian setiap parameter Algoritme Genetika dilakukan sebanyak 10 kali percobaan, untuk mendapatkan rata-rata yang terbaik. Berdasarkan pengujian yang telah dilakukan pada Bab 6, hasil pengujian parameter yang mendekati optimal didapatkan jumlah populasi sebesar 180, *cr*=0,9 dan *mr*=0,1, serta jumlah generasi sebesar 325 dengan nilai *fitness* terbaik didapatkan pada nilai 8,6870.
2. Dengan menggunakan Algoritme Genetika di dalam perhitungan FIS Tsukamoto terbukti dapat memberikan solusi yang lebih optimal dibandingkan dengan menghitung hanya dengan FIS Tsukamoto saja. Pada peramalan harga emas, nilai *error* yang dihasilkan oleh GA-FIS terkecil adalah sebesar 12,5801. Sedangkan nilai *error* yang dihasilkan oleh FIS saja adalah sebesar 13,3611. Hal ini disebabkan karena fungsi keanggotaan yang dihasilkan oleh GA didapatkan berdasarkan data-data sebelumnya, sehingga fungsi keanggotaan dapat beradaptasi dan mengurangi ketergantungan terhadap pakar.

7.2 Saran

Berdasarkan hasil dari keseluruhan penelitian yang telah dilakukan ini, peneliti memberikan saran untuk penelitian selanjutnya. Adapun saran yang dapat peneliti berikan adalah sebagai berikut:

1. Pada penelitian ini, optimasi dilakukan pada fungsi keanggotaan *fuzzy* saja. Pada penelitian selanjutnya, diharapkan dapat mengoptimasi *rule base* atau basis aturan *fuzzy*-nya juga sehingga hasil akhir yang didapatkan menjadi lebih optimal.
2. Dapat menggunakan metode *crossover*, mutasi, dan seleksi yang lain agar hasil akhir yang dikeluarkan lebih variatif dan dapat menghasilkan hasil akhir yang lebih optimal.
3. Dapat mengganti atau memodifikasi Algoritme Genetika dan FIS Tsukamoto dengan algoritme lainnya. Karena nilai *error* yang dihasilkan pada penelitian ini

memiliki selisih yang tidak begitu jauh dengan nilai *error* sebelum dioptimasi. Dengan mengkombinasikan 2 metode yang lain diharapkan dapat memberikan hasil yang lebih baik dan nilai peramalan dengan nilai *error* yang lebih rendah.



DAFTAR PUSTAKA

- Apriyanti, M., 2012. *Anti Rugi dengan Berinvestasi Emas*. Yogyakarta: Pustaka Baru Press.
- Armanda, R. S., 2016. Penerapan Algoritma Genetika Untuk Penentuan Batasan Fungsi Keanggotaan Fuzzy Tsukamoto pada Kasus Peramalan Permintaan Barang.
- Black, E. P., 2001. Dictionary of Algorithms, Data Structures, and Problems.
- Caraka, A. A., Haryanto, H., Kusumaningrum, D. P. & Astuti, S., 2015. Logika Fuzzy Menggunakan Metode Tsukamoto Untuk Prediksi Perilaku Konsumen di Toko Bangunan.
- Cristiono, D., 2005. *Aplikasi Pendukung Keputusan Dengan Menggunakan Logika Fuzzy*. Salatiga: Fakultas Teknologi Informasi, Universitas Kristen Satya Wacana.
- Desiani, A. & Arhami, M., 2006. *Konsep Kecerdasan Buatan*. Yogyakarta: Penerbit Andi.
- Dharmawati, A. & Aprilianto, H., 2014. Penerapan Metode Fuzzy Tsukamoto untuk Nilai Tukar Rupiah. *Jutisi*, 3(3), pp. 579-652.
- Evatiana, S., 2013. *Aplikasi Model Neuro Fuzzy Untuk Memprediksi Harga Emas*. Yogyakarta: FMIPA UNY.
- Fogarty, T., 1989. Varying the Probability of Mutation in Genetic Algorithm. *Proceeding of the Third International Conference on Genetic Algorithms*, pp. 104-109.
- Gen, M. & Cheng, R., 1997. *Genetics Algorithm & Engineering Design*. Hoboken: John Wiley & Sons, Inc..
- Gupta, R., Hammoudeh, S., Kim, W. J. & Simo-Kengne, B. D., 2014. Forecasting China's Foreign Exchange Reserves Using Dynamic Model Averaging. *The Role of Macroeconomic Fundamentals, Financial Stress and Economic Uncertainty*, Volume 28, pp. 170-189.
- Hadavandi, E., Ghanbari, A. & Abassian-Naghneh, S., 2010. *Developing a Time Series Model Based on Particle Swarm Optimization for Gold Price Forecasting*. s.l., 2010 Third International Conference on Bussiness Intelligence and Financial Engineering Developing.
- Halim, A., 2005. *Analisis Investasi*. Jakarta: Salemba Empat.
- Hanke, J. E. & Winchern, D. W., 2004. *Bussiness Forecasting*. 8th ed. United States of America : Pearson Education, Inc.
- Hasibuan, S., 2016. *Penerapan Sistem Fuzzy Untuk Prediksi Harga Kelapa Sawit*. Yogyakarta: FMIPA UNY.
- Hassani, H., Gupta, R., Silva, E. S. & Segnon, M. K., 2015. Forecasting the Price of Gold. *Applied Economics*, 47(39), pp. 4141-4152.

Husada, H., Cholissodin, I. & Bachtiar, F., 2018. Optimasi Penjadwalan Kuliah Pengganti Menggunakan Algoritme Genetika. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 2(9), pp. 2829-2834.

Hussein, S. F. M., Shah, M. B. N. & Jalal, M. R. A., 2011. Gold Price Prediction Using Radial Basis Function Neural Network. *Proceeding of the 4th International Conference of Modelling, Simulation and Applied Optimization (ICMSAO)*.

Indriani, P. & Harjahdi, 2013. Analisis Fundamental dan Teknikal dalam Pengambilan Keputusan Untuk Transaksi Emas di Perdagangan Berjangka. *Jurnal Akutansi dan Keuangan*, Volume 4, pp. 55-62.

Kusumadewi, F., 2014. *Peramalan Harga Emas Menggunakan Feedforward Neural Network dengan Algoritma Backpropagation*. Yogyakarta: FMIPA UNY.

Kusumadewi, S., 2003. *Artificial Intelligence (Teknik dan Aplikasinya)*. Yogyakarta: Graha Ilmu.

Kusumadewi, S. & Purnomo, H., 2004. *Aplikasi Logika Fuzzy Untuk Pendukung Keputusan*. Yogyakarta: Graha Ilmu.

Laribi, M. A., Milka, A., Romdhane, L. & Zeghloul, S., 2004. A Combined Generic Algorithm Fuzzy Logic Method (GA FL) in Mechanisms Synthesis. *Mechanism and Machine Theory*, Volume 39, pp. 717-735.

Lin, W. Y., Lee, W. Y. & Hong, T. P., 2003. Adapting Crossover and Mutation Rates in Genetic Algorithm. *Journal of Information Science and Engineering*, Volume 19.

Mahmudy, W. F., 2006. Optimasi Algoritma Genetika pada Optimasi Model Penugasan.

Mahmudy, W. F., 2014. *Optimisation of Integrated Multi-Period Production Planning and Scheduling Problems in Flexible Manufacturing Systems (FMS) Using Hybrid Genetic Algorithms*. South Australia: School of Engineering, University of South Australia.

Mahmudy, W. F., 2015. *Dasar-Dasar Algoritma Evolusi*. Malang: FILKOM UB.

Mahmudy, W. F., Marian, R. M. & Lee, L., 2013. Optimization of Part Type Selection and Loading Problem with Alternative Production Plans in Flexible Manufacturing System Using Hybrid Genetic Algorithms. *5th International Conference on Knowledge and Smart Technology (KST)*, 31-1 Jan-Feb, Volume 2, pp. 81-85.

Michalewicz, Z., 1996. *Genetic Algorithms + Data Structures = Evolution Programs*. 3rd ed. Charlotte: s.n.

Ni, M.-Y., 2006. *Stock Market Trading*. Worcester: Worcester Polytechnic Institute.

Pierdzioch, C., Risse, M. & Rohloff, S., 2014. The international Business Cycle and Gold-Price Fluctuations. *The Quarterly Review of Economics and Finance*, 54(2), pp. 292-35.

Rebizant, W., Szafran, J. & Wiszniewski, A., 2011. *Digital Signal Processing in Power System Protection and Control*. London: Springer Science & Business Media.

Ren, Y., Chen, X. & Logic, A. F., 2010. *Fuzzy Rule Representasion and Knowledge Base Construction in Expert System*. s.l.:2010 International CONference on Inteligent System and Knowledge Engineering (ISKE).

Shafiee, S. & Topal, E., 2010. *An Overview of Global Gold Market and Gold Price Forecasting*. s.l.:Resources Policy.

Sihananto, A. N. & Bachtiar, F. A., 2017. *Gold Price Movement Forecasting Using Hybrid ES-FIS*. s.l.:SIET.

Sudjana, 1996. *Metode Statistika*. Bandung: PT. Tarsito.

Suhartono, E., 2015. Optimasi Penjadwalan Mata Kuliah dengan Algoritma Genetika (Studi Kasus di AMIK JTC Semarang). *INFOKAM*.

Surya, M. N. A., Hidayat, N. & Rahayudi, B., 2019. Implementasi Fuzzy Inference System (FIS) Pada Metode Tsukamoto Dalam Peramalan Produksi Roti. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 3(1), pp. 7871-7877.

Sutojo, T., Mulyanto, E. & Suhartanto, V., 2011. *Kecerdasan Buatan*. Yogyakarta: ANDI.

Wibawanto, H., 1998. *Pengendali Berbasis Logika Kabur*. 14th ed. s.l.:Elektro Indonesia.

